Report TA 78-2-3

LEVEL

NOVEL ECCM TECHNIQUES FOR ARMY TACTICAL COMMUNICATIONS

R. L. Pickholtz
Telecommunications Associates
3613 Glenbrook Road
Fairfax, VA 22031

29 June 1979

Final Report for Period January-June 1979

~~For Internal CORADCOM Distribution~~

Prepared for

AD A093882

DDC FILE COPY

81 1 15 056

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER (14) TA-79-2-3 2. GOVT ACCESSION NO. AD-A093 883	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Novel ECCM Techniques for Army Tactical Communications	5. TYPE OF REPORT & PERIOD COVERED
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Raymond L. Pickholtz Herman Helgert Roger Lang R., Lawrence Milstein Ray Pettit R. (15)	8. CONTRACT OR GRANT NUMBER(s) DAAB07-78-C-0174
9. PERFORMING ORGANIZATION NAME AND ADDRESS Telecommunications Associates 3613 Glenbrook Road, Fairfax, VA 22031	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS U.S. Army CORADCOM ATTN: DRDCO-COM-RM-2 Fort Monmouth, N.J. 07703	12. REPORT DATE 29 June 1979
	13. NUMBER OF PAGES 242
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)	15. SECURITY CLASS. (of this report) Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

CORADCOM Internal

DISTRIBUTION STATEMENT A

Approved for public release;

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

ECCM, spread spectrum, frequency hopping, adaptive traffic control, adaptive null steering antennas, convolutional codes, tactical communications.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This Report contains the basic analysis for certain novel techniques in ECCM which are suitable for tactical communications. Included are analyses for adaptive capacity allocation, low rate coding, adaptive antenna arrays, multichannel synchronization and frequency hopping.

DD FORM 1473   EDITION OF 1 NOV 65 IS OBSOLETE

NOVEL ECCM TECHNIQUES

FOR

ARMY TACTICAL COMMUNICATIONS

JUNE 1979

CONTRIBUTORS

CONTRIBUTORS TO THIS REPORT ARE: (IN ALPHABETICAL ORDER)

HERMANN HELGERT, PhD

ROGER LANG, PhD

LAURENCE MILSTEIN, PhD

RAY PETIT, PhD

RAYMOND PICKHOLTZ, PhD

TABLE OF CONTENTS

ILLUSTRATIONS

ILLUSTRATIONS
(Continued)

ILLUSTRATIONS
(Continued)   <u>Page</u>

TABLES

CHAPTER 1

Introduction And Executive Summary

This document is the final report on novel techniques for reducing the ECM vulnerability of tactical communications. Tactical communications have been plagued by having had so many constraints placed on them that there has, in the past, been little attention paid to the vulnerability of field systems to electronic countermeasures. Army tactical communications are applied in a wide variety of circumstances and hostile environments. The systems employed include field radio HF, VHF and UHF single channel and multichannel and SHF satellite communications both fixed and mobile. In addition, there are line of sight microwave radio systems and UHF tropospheric scatter over the horizon. The deployment of various tactical field radios is prodigious and each system must frequently operate in a grid network tied to a backbone transmission system and switching centers. The application on the tactical communication is widely varied and includes voice, data, message store and forward. Any or all of the information may be analog or digital and may or may not be encrypted.

Vulnerability to jamming of such tactical communication systems has been recognized for some time, but because of conflicting restraints cost, and the apparent unavailability of a sufficiently small, inexpensive, and reliable technology to implement ECCM techniques, these techniques have lagged behind.

ECCM techniques come in many different forms and they in-

clude various modes of spread spectrum (PN direct sequence, frequency and time hopping, chirp, etc.), signal cancellers, diversity, coding, hall steering antennas, power control, and transmission coordination among the favorites. In this study, where we concentrate on multichannel digital communications we address some ideas which would appear to have the greatest possible impact on ECCM effectiveness given, the constraints imposed by the tactical scenario and even more specifically by the nature of tactical multichannel communications. These constraints form a checklist of issues in tactical multichannel communications which guided our thinking and they are:

* Small, stationary, isolated transmitters

* Relatively low cost

* Limited bandwidth availability and preassigned frequency allocation

* The quality of transmission may be allowed to depend on jamming conditions and on the priority class of the user

* The availability of channels and the concomitant grade of service (G.O.S.) may be allowed to depend on jamming conditions and on the priority class of the user

* Highly sophisticated jammers are unlikely (it could be easier to destroy the antenna tower)

* There may exist a geometric distribution of both friendly and unfriendly interference.

This checklist, while formidable in the limitations that it imposes on what is achievable, does nevertheless narrow

the focus on those ECCM techniques which stand a chance for
success. The bright side of the picture is the anticipation
that VHSI and VLSI technique which will be available by 1985
will allow the practical implementation of many of the novel
ideas which today would appear too sophisticated. With this
in mind we have chosen for investigation six ideas which appear
to be promising. They are:

* Channel monitoring and adaptive channel capacity
allocation (Chapters 2 and 3)

* Means for protecting the synchronization of multi-
channel digital communications (Chapter 4)

* The use of low rate block and convolutional codes in
lieu of traditional spread spectrum (Chapter 5)

* Adaptive null-steering antenna arrays (Chapter 6)

* Bandwidth reduction with multiple alphabet signalling
(Chapter 7)

* Frequency Hopping systems (Appendix A).

The following is a brief synopsis of each Chapter of this
report.

Chapter 2 - Bit Error Rate Monitoring Using
Extreme Value Theory

In order to be able to react to a jamming condition, it
is necessary to monitor the channel. In a digital system the
bit error rate (BER) is the most logical and directly signif-
icant variable to observe. Unfortunately, a working system
requires a very low BER ($< 10^{-5}$) so that long measurement

times are often necessary to obtain an estimate of BER with
any reasonable confidence. Now since the signal and noise
structure is known, it is possible to obtain estimates using
extreme value theory (EVT) more directly and more quickly (with
fewer measurements than would appear to be necessary). The
monitoring technique described here is relatively simple to
implement and is directly amenable for use in conjuction with
the dynamic channel allocation scheme described in Chapter 3.

Chapter 3 - Dynamic Allocation Of Channel Capacity

Given that we can monitor the condition of the channel and
determine whether or not a jammer is present, we could, ordi-
narily go into a spread spectrum mode to provide processing
gain (PG) and corresponding jam protection for communications.
However, since most tactical radios are bandwidth limited and
tend to use the available bandwidth to provide 6, 12, or 24
or more voice and data channels, it is not possible to merely
spread the emissions without some sacrifice. One way to ensure
performance and protection for an important class of users is
to use the bandwidth of possibly less important users during
an attack. This need not mean that the less important users
are denied access to communications but rather that their grade
of service (G.O.S.) may be lowered by increasing the probability
that they will be denied access to the service (depending on the
total demand). For example, while in a non-stressed situation
all users might have access to all channels, under stress
priority class #1 might be assigned 80-100% of the channel

- 4 -

capacity while a lower priority class might have access to only say 40% of the channel resources. The question is then how does one make the assignment to keep some measure of overall performance at a satisfactory level. In Section 2 we show a method for making the allocation so that high priority users maintain their GOS under jamming while lower priority users still have reasonable, albeit reduced access to communications. Specific examples illustrate the method. We believe the method can be made adaptive in conjunction with the channel monitoring scheme proposed in Chapter 2.

Chapter 4 - Susceptibility Analysis Of A Frame Synchronization Technique For Multichannel Frequency-Hopped BNCFSK Systems In Partial-Band Noise Or CW-Tone Jamming

Perhaps the most vulnerable aspect of a multichannel time division multiplexed digital signal is the mechanism and signal format used for frame synchronization. If frame synchronization is lost or otherwise interferred with then the bit stream becomes useless as an information signal. In this Chapter we investigate the effect of various partial band jamming signals on the frame synchronization. The analysis is presented in a general form in order to permit parametric studies for optimizing the frame synchronization anti-jam technique. Several numerical examples are worked out of the given formulas in order to show the effect of noise jamming without frequency hopping, noise jamming and partial hard jamming with frequency hopping and various signal to jam ratios on the probability of correct frame synchronization. It would appear that perhaps

greater protection ought to be given to the frame synch signal than has hitherto been the case. Several coding schemes are possible candidates.

Chapter 5 - Low Rate Error Correcting Codes
As An Alternative To Conventional Spread Spectrum

Since direct sequence spread spectrum derives its processing gain by integrating over many DS chips per data bit, the modulated DS signal can be viewed as a low rate code. The DS modulation however is not exploited as a code since none of its structural properties (except autocorrelation) is exploited. Indeed DS pseudo-noise can be viewed as an "inner" channel operation which is undone by the receiver correlation and quite independent of any data that is sent. As such it is an "applique" and we should be able to do better, at the same information rate; by coding. In this Chapter we propose a particular class of low rate linear block codes because 1) they have the best distance properties for their length and 2) they are amenable to effective simple majority logic decoding. We show the construction of both encoder and decoder and present an analysis of the performance of these codes. Because of the code structure, we can not only <u>correct</u> errors and thereby improve the bit error rate (BER) for a given spread spectrum over conventional methods (in the range of interesting E_b/N_o) but we can also <u>detect</u> errors which allow us either to <u>retransmit</u> or to <u>delete</u> blocks with many more errors than we can correct! There are many situations especially in the transmission of digitized speech when it is preferable to

delete error blocks than accept them. Thus the several perfor-
mance measures are analyzed. They are bit error rate (BER),
undetected bit error rate (UBER), and data bit deletion rate
(DBDR). The analysis prompted numerical investigations by
computer and asymptotic analysis to determine the precise range
where coding gain (over spread spectrum) is achieved. In the
range of E_b/N_o of 10 dB or greater for hard decision decoding,
significant coding gain is achieved. The BER can be reduced by
orders of magnitude below that achievable by straight spread
spectrum. The UBER and DBDR are even more impressive. These
codes also lend themselves to soft decision decoding when coding
gains of 10 dB or more are theoretically possible down to 3 dB.

Further in this chapter, we propose a new class of very low
rate convolutional codes which we call "Complete Convolutional
Codes." These codes are characterized by the property than for
constraint length K the generators consist of all 2^{k-1} binary
polynomials (all tap connections) and hence form a vector space.
These codes have demonstratable good performance and coding gain
since we show that the free distance of the codes can be explicitly
derived and an upper bound on BER can be computed from this free
distance.

Significant advantages can be had by using coding as an
alternative or in conjunction with spread spectrum. These advantages
are summarized in the chapter.

Chapter 6 - Adaptive Null Steering Antenna Arrays

Perhaps the most promising of ECCM techniques available in
a bandwidth limited environment is that of null steering antennas.
It may be possible to achieve 40 or 50 dB of jamming margin with
hardly any substantial bandwidth expenditure. Furthermore,
tactical multichannel radio communications usually exist on
stationary sites and do not have to operate on rapid angular
dynamics (as in a plane) so that perhaps costs can be manageable.
The problem in adaptive arrays however, is that they may be
vulnerable to "blinking" jammers matched in period to the natural
modes of the array loops. In this section, a basic analysis,
starting with first principles is made for adaptive antenna
arrays. Both the linear mean square (LMS) or Widrow algorithm
and the maximum signal to noise (SNR) or Applebaum algorithm are
considered. The important result is that we can determine the
exact deterministic transient solution and periodic solution of
the differential equations of the array. We believe that this
has never been done previously and that this will have an enormous
impact on the understanding of the vulnerabilities and design of
adaptive arrays. In particular, it should shed light on the effects
of "blinking" jammers (a subject which has recently received much
attention by experimenters). Furthermore, the ability to solve
the equations explicitly for deterministic signals allows us
fortuitously to obtain the analytic behavior of the adaptive array
under stochastic or noisy jamming signals. The pursuit of this

approach should give us a fairly complete picture of the effects of various jammers on adaptive arrays and the means to make them more effective.

Chapter 7 - Modulation and Spreading Techniques

This chapter presents the results of an extensive effort to discover the optimum multialphabet modulation scheme to be used with spread spectrum that will result in the "best" combination modulation/spreading with respect to a jammer under the constraint of limited total bandwidth. Various modulation schemes were considered including BPSK, QPSK, 16-ary QASK, FSK and the effects that a number of jammers would have on them. It appears that QPSK for direct sequence on how coherent FSK with coding in a frequency hopped system is best. The curves in this chapter can be used to demonstrate the trade-off possible by using adaptive and dynamic channel allocation as discussed in chapter 3.

Chapter 8 - Conclusions and Recommendations

This very brief chapter is included to provide an overview of which has been accomplished, how and whether some of the ideas that evolved during the course of this work ought to be implemented, and some recommendations for further work both in analysis and in development.

Appendix A - Spread Spectrum Background
Frequency Hopping

In this appendix we provide an up to date and thorough review of frequency hopping. This appendix together with those of a companion report by S consultants are a self-contained exposition of spread spectrum, its important features, problems and limitations. Several of the topics in this appendix have not appeared in the literature before. The treatment here includes the properties of frequency hopping, correlation, effects of partial band jamming, the use of coding, acquisition, digital frequency synthesiser and a comparitive analysis of frequency hopping and direct sequences modulation.

CHAPTER 2

Bit Error Rate Monitoring Using Extreme-Value Theory

The general technique and rationale for using extreme-value theory (EVT) as the basis for a bit error rate (BER) performance monitor is described in [2.1]. An appendix is included at the end of this section which contains extracts from [2.1] and hence it provides the basis for the remainder of this section. The specific manner in which it will be used in the current study will be described below.

As can be seen from either [2.1] or the Appendix, each of the two BER estimates (i.e. one which estimates the number of +1's in error and one which estimates the number of -1's in error) depends upon two parameters. One parameter (labeled u) is a function of both the location of the initial distribution as well as the spread or the standard deviation of the distribution of the underlying samples, while the second parameter (labeled α) is inversely proportional to the standard deviation of the initial distribution. When estimating the number of -1's in error one is interested in the right hand tail of the density and hence, in the notation of [2.1]-[].5], in the parameters u_n and α_n. Analogously, the parameters u_1 and α_1 are appropriate for estimating the left-hand tail of a density and hence are used to estimate the number of +1's in error.

If it is assumed that we are dealing with a coherent QPSK system and that we are estimating the BER in one of the two biphase channels, then for equally likely input data, the

number of +1's in error should, on the average, equal the number
of -1's in error. Also, from the defining equations for u_n,
u_1, α_1 in the Appendix, it can be seen that for this situation,

$$u_n = -u_1 \tag{2.1}$$

and

$$\alpha_n = \alpha_1 \tag{2.2}$$

Using these two equations, it is reasonably straightforward to
determine when a jammer is present. Consider, for example, a tone
jammer, the effect of which is to cause a shift in the probability
density of the final test statistic. Figure 2.1 illustrates a
typical situation. Figure 2.1a shows the density functions when
the jammer is not present. Figure 2.1b shows the effect of the
jammer assuming the shift (due to the jammer) is to the right.
It is clear that many more +1 decisions will be made than -1
decisions, hence the existence of an outside disturbance (e.g.
a jammer) has been confirmed. Alternately, $u_1 >> |u_n|$, thus vio-
lating Eq. (2.1).

If instead of being a tone jammer the interference had been
a noise jammer, the situation would look as shown in Figure 2.1c.
Now the number of +1 decisions is still equal to the number of
-1 decisions, but because the variance of either density is much
larger than what it initially was, the values of α_1 or α_n will be
much smaller, thereby indicating the presence of an outside dis-
turbance. This latter statement assumes that reasonable esti-
mates of α_1 and α_n, say $\hat{\alpha}_1$ and $\hat{\alpha}_n$, respectively, can be
obtained when the jammer is present and

(a) No jamming

(b) Tone Jammer

(c) Noise jammer
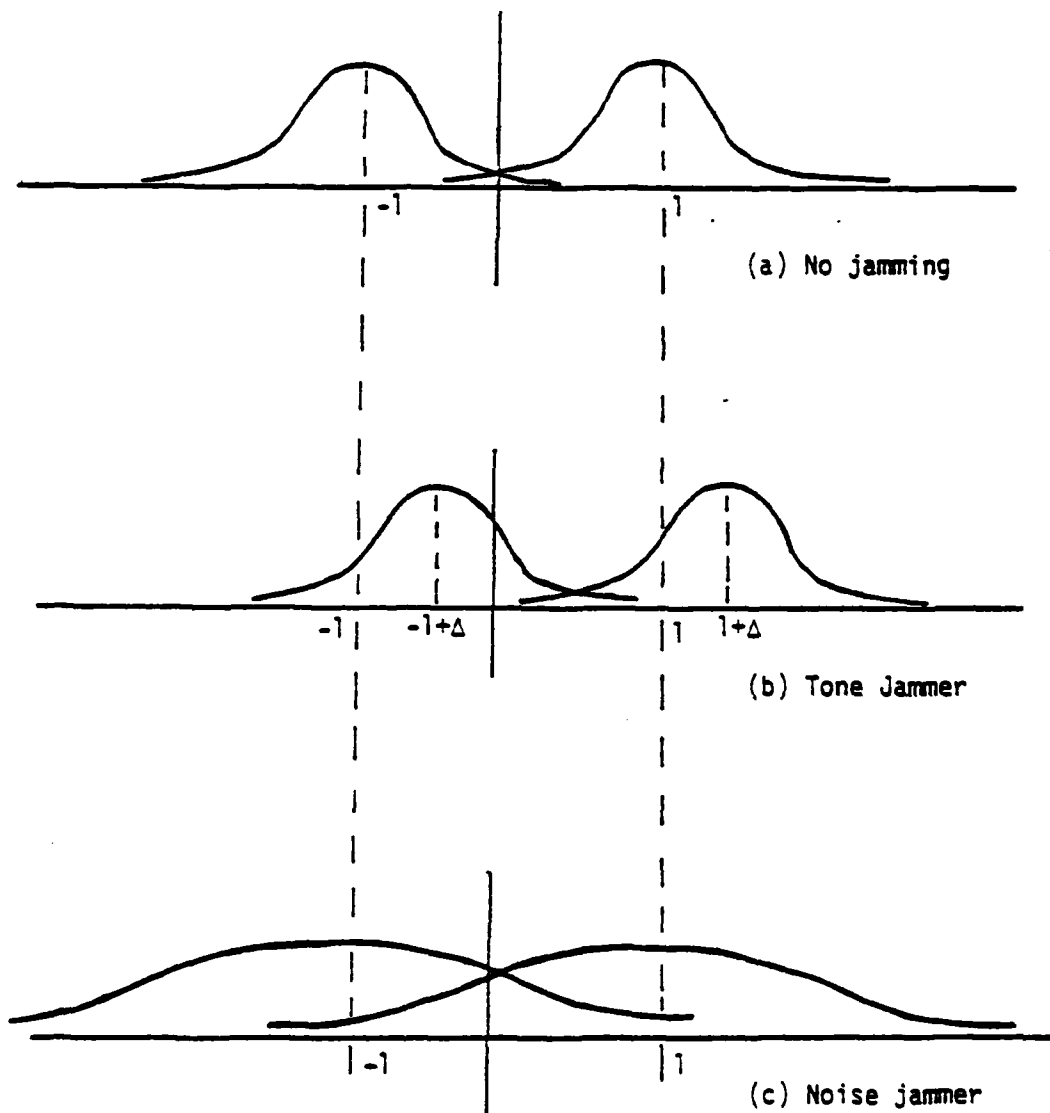
Figure 2.1   Qualitative Effect of Jamming

before any counter-measures have been taken to combat the jammer. This will only be the case if a learning period is provided during which time the receiver knows whether a given sample is a $=1$.

If this learning period is not provided, one can still get an indication of the jammer's presence in the following manner: As described in[1], the parameters c_1, c_n, u_1, and u_n are all estimated using the extremes (either maxima or minima) of sets of n samples. When the learning period is absent, some of those extremes will be incorrect, and in particular, when the noise jammer is present, most of these samples will be incorrect. However, they will be incorrect in a very specific manner, namely they will all be very close to the threshold used to distinguish between ± 1's (zero in this case). This will result in a very small standard deviation and hence a very large value of $\hat{a}$. Hence even though an incorrect value of $\hat{a}$ is being obtained, it will nonetheless be different enough from the value of $\hat{a}$ obtained when the jammer was absent to clearly indicate the jammer's presence.

Having determined the fact that the system is being jammed, it remains to indicate what response to the jammer the system should initiate. Ideally, one would like to use just enough counter-measure (e.g. spectrum spreading) to bring the system back to where it was (in a BER sense) before it was jammed. This, however, requires precise knowledge of the BER while the system is being jammed and before any counter-measure is initiated. Because EVT is only accurate on the tail of a

density, and because the system at this point is no longer operating on the tail, accurate BER estimation using EVT cannot take place. Therefore, a compromise is to spread the spectrum some predetermined amount, say 50% of the total possible spread, and test to see if this puts the system back on the tails (e.g. test to see if the estimated parameters $\hat{\alpha}_1$, $\hat{\alpha}_n$, $\hat{\delta}_1$, and $\hat{\delta}_n$ return to their pre-jamming values). If it does, operate the system as it is, if it does not, spread the spectrum the remaining 50%. Assuming the system has at its disposal enough processing gain to nullify the jammer, the system will at one of the two steps above return to operating at a low BER, and accurate BER estimation can then once again take place.

To verify the above procedure essentially requires a computer simulation. However, some indication of how it will perform can be obtained by assuming a specific system and using the true parameter values whenever they are needed. Therefore assume we are looking at the inphase channel of a QPSK system (or equivalently a simple BPSK system) consisting of a coherent demodulator followed by an integrate and dump filter. The signal is received in the presence of additive white gaussian noise (AWGN) of two-sided noise spectral density $\frac{No}{2}$ as well as the (potential) presence of a gaussian noise jammer occupying a bandwidth equal to that of the main lobe of the signal spectrum.

There are two considerations that must be addressed in evaluating the error-rate monitoring procedure for this situation. Since, as was indicated above, for the noise jammer the

- 15 -

key parameters are α_1 and α_n, it has to be ascertained that these values do indeed undergo a significant enough change in value when a jammer is present (from what they were when the jammer was absent), and that given a sizable enough change in value, this change can be estimated from samples of the detected waveform.

Consider the first question. Because both the noise and interference are gaussian, the output statistic of the receiver has a symmetrical probability density. Under this condition, it is straightforward to show that $\alpha_1 = \alpha_n$. Therefore only values of α_1 need be considered. For any density, α_1 is defined by

$$\alpha_1 = nf(u_1) \tag{2.3}$$

where u_1 is the solution of the following equation (see the Appendix):

$$F(u_1) = \frac{1}{n} \tag{2.4}$$

In (2.3) and (2.4), $F(\cdot)$ and $f(\cdot)$ are the cumulative distribution function and probability density function, respectively, of the output statistic and the significance of n is explained in [2.1]-[2.5] and will be seen below. If the signal component of the output statistic for the problem at hand is normalized to unity, it can be shown in a straightforward manner that the following equation holds:

$$F(u_1) = \Phi \left[\frac{u_1 - 1}{\left[\frac{N_o}{2E} + \frac{J}{2S} \frac{1}{2E_c} \right]^{1/2}} \right] \tag{2.5}$$

where E is the signal energy, S the signal power, J the jammer power, 2fc the bandwidth in Hertz of the mainlobe of the signal spectrum, T the duration in seconds of one data symbol, and

$$\phi(x) \triangleq \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\frac{y^2}{2}} \, dy \tag{2.6}$$

Example

Assume now that the system is operating with an E/No of 10 dB and without any spread spectrum processing gain. With no jammer present, α_1 can be shown to equal 14.6 when $n = 10^3$. If a jammer with 30 dB more power than the signal (i.e. $\frac{J}{S} = 30$ dB) in a bandwidth of $2fc = \frac{2}{T}$ suddenly starts to interfere with signal, the value of α_1 reduces drastically to 1.7×10^{-3}.

If the system uses this reduction in α_1 as an indication of the jammer's presence and hence spreads its spectrum by a factor of, say, 500 (i.e. 2fcT = 1000), α_1 will then increase to 4.4. This clearly indicates that while significant improvement in system performance has been obtained by spreading the spectrum, the spread was not sufficient to bring the system back to the point it was before the jamming signal was received. In any event, it is seen that α_1 does indeed undergo a significant enough change in value.

Relative to the second question, suppose we have a learning period available and that α_1 is to be estimated by $\hat{\alpha}_1$ defined as follows:

$$\hat{\alpha}_1 \triangleq \frac{\pi}{\sqrt{6}\,\hat{\sigma}_{min}} \tag{2.6}$$

where

$$\hat{\sigma}_{min} = \left\{ \frac{1}{N-1} \sum_{i=1}^{N} [X_{i_{min}} - \bar{X}_{min}] \right\}^{\frac{1}{2}} \tag{2.7}$$

is the estimate of the standard deviation of the minimum of n
samples of the output statistic and is obtained by computing
the sample standard deviation of N samples, each one being the
smallest from a group of n independent samples of the output
of the receiver. The rationale for the estimate of (2.6) is
described in [2.2] and [2.3]. Each of the $X_{i_{min}}$ in (2.7)
is the minimum of n independent samples, and $\bar{X}_{min}$ is the sample
mean of the N minima.

Obtaining the exact probability density of $\hat{a}$, does not
appear to be tractable. Therefore, an approximation which
becomes more and more valid is N increases will be used. In
[2.2] and [2.3], it is shown that $(\hat{a}_1)^{-1}$ is asymptotically nor-
mal with mean equal to $1/a_1$ and variance equal to $1.1/N_1^2$. There-
fore, using this asymptotic result, we obtain a confidence inter-
val on $\hat{a}$, as follows:

$$P\left\{ a\alpha_1 \le \hat{a}_1 \le b\alpha_1 \right\} = \phi\left(\frac{\frac{1}{a}-1}{\sqrt{\frac{1.1}{N}}} \right) - \phi\left(\frac{\frac{1}{b}-1}{\sqrt{\frac{1.1}{N}}} \right) \tag{2.8}$$

where a and b are constants and $\phi(\cdot)$ is defined in (2.6).

If, for example, we are interested in the probability of
a_1 being within a factor of two of its true value when N=20,
then

$$P\left\{\frac{\alpha_1}{2} \leq \hat{\alpha}_1 \leq 2\alpha_1\right\} = .985$$

Therefore, notwithstanding the fact that we are using an asymptotic result for finite N (e.g. N = 20), it appears that one can indeed obtain an accurate enough estimate of α_1 to determine when the system is being jammed.

Conclusions

From the above results, it is clear that EVT can serve both to indicate the presence of a jammer as well as to estimate the average probability of error of the system once a sufficient degree of processing gain is given the system to combat the jammer. Furthermore, the actual implementation of such an error rate monitor is relatively straightforward, requiring only that extreme samples taken from sets of samples of the final receiver test statistic be accumulated and then used to compute the estimates of the u's and the α's (e.g. Eq. (2.7)). From these estimates, the desired estimates of probability of error as given by Eqs. (A.1) and (A.2) of the Appendix are simply obtained.

Appendix to Chapter 2

Review of Extreme-Value Theory

Given a collection of n iid random variables, there are
certain conditions on the distribution function of the random
variables such that the distribution of the largest random
variable from the set of n can be expressed in an asymptotic
form (see, e.g. [2.3]-[2.5]). Specifically, Gumbel in [2.3]
shows there are three distinct asymptotic forms. Of the three,
the so-called "exponential type" appears to be of most interest.
This limiting form (for maximum values) applies for any initial
distribution which goes to zero on its right-hand tail in a
manner similar to an exponential. In particular, it is shown
in the above references that, with $F(x)$ and $f(x)$ the initial
distribution and density functions, respectively, if

$$\lim_{x \to \infty} \frac{f(x)}{1-F(x)} = -\lim_{x \to \infty} \frac{f'(x)}{f(x)}$$

then the distribution function of the maximum value is given by

$$F_{max}(x) = \exp\left[-\exp[-\alpha_n(x-u_n)]\right]$$

where

$$F(u_n) = 1 - \frac{1}{n}$$

and

$$\alpha_n = nf(u_n)$$

The analogous condition for the left-hand tail (i.e., minimum value) is as follows. If

$$\lim_{x \to \infty} \frac{f(x)}{F(x)} = \lim_{x \to -\infty} \frac{f'(x)}{f(x)}$$

then

$$F_{min}(x) = 1 - \exp\left[-\exp[\alpha_1(x-u_1)]\right]$$

where

$$F(u_1) = \frac{1}{n}$$

and

$$\alpha_1 = nf(u_1)$$

Since, in general, the underlying initial distribution is unknown, the parameters u_n, α_n, u_1, and α_1 have to be estimated from sample values. This usually proceeds as follows. A total of $K = nN$ samples from a given distribution are taken and divided into N groups of n samples per group. If it is desired to estimate u_n and α_n, in each of the N groups the largest of the n samples is chosen. From these N largest values, it can be shown that a variety of procedures exist from which u_n and α_n can be estimated. A method for obtaining maximum-likelihood estimates of u_n and α_n is described in (2.3) along with various other estimation techniques.

It is desired to estimate u_1 and α_1, a similar procedure using minimum instead of maximum values can be performed. Finally, appropriate estimates for the two types of error in a binary

digital communication system are given by

$$P_{fa} = \frac{1}{n} \exp[-\alpha_n (x_t - u_n)] \qquad (A.1)$$

and

$$P_{fd} = \frac{1}{n} \exp[\alpha_1 (x_t - u_1)] \qquad (A.2)$$

where x_t is the threshold that the system is operating with (e.g. zero for a BPSK system) and where the two probabilities have been labeled P_{fa} and P_{fd} in analogy with classical radar notation, to represent the false alarm and false detection probability respectively.

References to Chapter 2

[2.1]   L.B. Milstein, "Performance Monitoring of Digital
        Communication Systems Using Extreme-Value Theory,"
        IEEE Trans. Comm., Vol. COM-24, No. 9, pp. 1032-1036,
        September 1976.

[2.2]   L.B. Milstein, D.L. Schilling, and J.K. Wolf, "Robust
        Detection Using Extreme-Value Theory," IEEE Trans. Inform.
        Theory, Vol. IT-15, pp. 370-375, May 1969.

[2.3]   E.J. Gumbel, Statistics of Extremes. New York: Colum-
        bia Univ. Press, 1958.

[2.4]   , "Statistical Theory of Extreme Values and Some
        Practical Applications," NBS Appl, Math. Ser. 33, U.S.
        Government Printing Office, Washington, D.C.

[2.5]   B. Gnedenko, "Sur la Distribution Limite du Terme Maxi-
        mum d'Une Aleatoire," Ann. Math., Vol. 44, pp. 423-453,
        July 1943.

CHAPTER 3

Dynamic Allocation of Channel Capacity

In the Interim Report for the period September 30, 1978
to January 31, 1979, a specific technique for the dynamic allo-
cation of channel capacity was presented. This technique made
use of an equivalent traffic model in which users from different
priority classes shared a certain percentage of the total
resources available, and were then credited with a certain
"effective" traffic which was in general greater than their
actual traffic. This increase in traffic was of course due to
the fact that the channels assigned to a given user class, say
the ith class, were in general available to other user classes
and hence affected the grade of service (GOS) (i.e. the proba-
bility that an incoming call will be blocked) of a user in the
i^{th} class.

However, further analysis of that model demonstrated that
it had a significant flaw to it. This can be demonstrated as
follows: Assume a channel allocation between two user classes
is as shown in either Table 3.1 or Table 3.2 below.

Table 3.1

user # \\ channel #	1	2	3 . . . . . n_1	n_1+1	n_1+2. C
1	x	x	x x	o	o . . . . o
2	o	o	o 0	x	x x

Table 3.2

channel #user #	1	2	3 . . . . n_1	n_1+1	n_1+2 C
1	x	x	x . . . . o	x	x . . . . o
2	o	o	o . . . . o	x	x x

In both allocation schemes, there are a total of C channels. In the first scheme, user class #1 has n_1 channels fully dedicated to it, while in the second scheme, user class #1 has access to n_1+1 channels, n_1 on a fully dedicated basis, and one channel (the n_1+1th) on a shared basis with user class #2. For a given offered traffic of user class #1, say e_1 erlangs, it is physically obvious that the GOS_1 of user class #1 with the second allocation scheme has to be better than the GOS_1 with the first allocation scheme. Yet because the GOS_1 of user class #1 was computed in the Interim Report based upon the "effective" traffic, which was a linear combination of the actual traffic of user classes #1 and #2, it is clear that for some value of e_2 (and all higher values), the actual (i.e. offered) traffic of user class #2, the GOS_1 of the second allocation scheme will appear to be poorer than the GOS_1 of the first allocation scheme.

In other words, the flaw with the traffic model presented in the Interim Report was the manner in which the shared traffic was combined to yield the "effective" traffic, this latter quantity being the basis for the GOS calculations. Therefore, for this present report, it was necessary to use a different traffic model, and such a model will be presented below. While

the original motivation for changing to this new model was
to avoid the flaw just described in the old model, it turned
out that this new model had the further advantage of being easier
to use. In particular, whereas the first model required the
computation of the pseudo-inverse of an NxN matrix, N being the
number of priority classes, no such calculation is necessary
with the current technique. In what follows, the model will
first be described, and then numerical results illustrating its
behavior will be presented.

Assume that there are three basic priority classes with n_i
users in the ith class, $i=1,2,3$, and let

$$n \stackrel{\Delta}{=} \sum_{i=1}^{3} n_i \tag{3.1}$$

be the total number of users. Let e_i be the number of erlangs
of traffic of the ith priority class, and let

$$e \stackrel{\Delta}{=} \sum_{i=1}^{3} e_i \tag{3.2}$$

While there are many types of priority assignments that
could be made, the following one will arbitrarily be considered
in the remainder of this section: User class #1 will have highest
priority, followed by #2 and then #3. All users from all classes
will have access to every channel. However, if a user from class
#1 is blocked when attempting to access one of the channels,
that user will be allowed to "bump" one of the #3 users (pro-
viding at least one of the channels is in fact occupied by a
#3 user) and hence gain immediate access to the channel.

Given the above model, the grade-of-service (GOS) for a user

in each of the three priority classes will be computed. Assume
there are a total of c channels, where in general c<n and in
most instances c<<n, and denote by $P(c,e)$ the probability of
blocking in a system with c channels and e erlangs of traffic
as computed from the so-called Erlang-B model.

Consider user class #1. A call will be blocked in this
case only if all c channels are occupied by either other #1
users or #2 users. To compute this probability, consider the
probability of one channel, say the jth channel, being occupied
by a user from class #i. Denoting this latter probability
by P_i, we have

P_i = P[channel j occupied by a user form #i | channel j is
occupied] · P[channel j is occupied].

The conditional probability on the rhs will be taken to be n_i/n.
The second term on the rhs is computed as follows:

$$P[\text{channel j is occupied}] = \sum_{k=0}^{c} P[\text{channel j is occupied}|$$

exactly k channels occupied] · P[exactly k channels occupied]

$$= \sum_{k=0}^{c} \frac{k}{c} \frac{\frac{e^k}{k!}}{\sum_{\ell=0}^{c} \frac{e^\ell}{\ell!}}$$

Therefore

$$P_i = \frac{n_i}{cn} \sum_{k=0}^{c} \frac{\frac{e^k}{(n-1)!}}{\sum_{\ell=0}^{c} \frac{e^\ell}{\ell!}} \tag{3.5}$$

Using (3.5), there is straightforward to show that

$P(i,j,k,\ell) \triangleq P$ {i channels used by #1, j channels used by #2,

k channels used by #3

ℓ channels not being used}

$$= \frac{c!}{i!j!k!\ell!} P_1{}^i P_2{}^j P_3{}^k (1-P_1-P_2-P_3)^\ell \qquad (3.4)$$

where $i+j+k+\ell = c$. Finally, GOS_1, the GOS of a user in class #1 is given by

$$GOS_1 = P(c,e_1+e_2)P_c \qquad (3.5)$$

where

$$P_c \triangleq \sum_{\substack{i=0 \\ i+j=c}}^{c} \sum_{j=0}^{c} \frac{c!}{i!j!} P(i,j,o,o) \qquad (3.\dot{}6)$$

and where $P(c_1 e_1+e_2)$ is the Erlang B blocking formula for c channels and e_1+e_2 erlangs of ordered traffic.

For a user in class #2, the GOS is simply

$$GOS_2 = P(c,e_1+e_2+e_3) = P(c,e) \qquad (3.7)$$

Finally, for a user in class #3, the definition of GOS has to be generalized somewhat. Since a #3 user can be "bumped" from a channel after he has started sending a message by a #1 user, the GOS of a #3 use will be defined as the probability of either being blocked or of being "bumped." Since these are mutually exclusive events, we have

$$GOS_3 = P(block) + P(bump) \qquad (3.8)$$

The last term on the rhs of (3.8) is the probability that a
user from class #1 wants to access a channel during the time that
a user from class #3 is transmitting and that no free channel
is available. To upper bound this probability, assume a #3
message uses one channel during the interval (o,t), where t
is a random variable with an exponential probability density.
The probability that at least one user from class #1 wants to
access the channel during that t second interval is given by

$$P\{\text{at least one \#1 arrives in } (o,t)\,|\,t\} = 1-e^{-\lambda_1 t} \qquad (3.9)$$

where λ_1 is the average number of arrivals/sec for user class #1.
Eq. (3.9) followed because of the assumed Poisson arrival density
for any of the users. To obtain the unconditional probability
that a #1 user arrives while the #3 is transmitting it is necessary
to average (3.9) over the density of t. Therefore

P{at least one #1 arrives while the #3 user is on}

$$= \int_0^\infty [1-e^{-\lambda_1 t}]\frac{1}{\mu} e^{-t/\mu} dt = 1 - \frac{\frac{1}{\mu}}{\frac{1}{\mu}+\lambda_1} = \frac{\lambda_1\mu}{1+\lambda_1\mu} = \frac{e_1}{1+e_1} \qquad (3.10)$$

where 1/μ is the average duration of a message, assumed the
same for any user in any priority class. Using (3.10), we have
P(bump) = P{(the #3 user is not initially blocked),

        (a #1 user arrives while the #3 user is on),

        (all channels are occupied when the #1 user arrives),

        (of all the #3 users using channels when the #1 user

          arrives, this particular #3 is the one bumped)}

- 30 -

$\leq$ P{user #3 not initially blocked, a #1 user arrives while #3

 user is on}

$$= [1-P(c,e)]\frac{e_1}{1+e_1}$$

Therefore, combining (3.8) and (3.11) yields

$$GOS_3 \leq P(c,e) + [1-P(c,e)] \frac{e_1}{1+e_1}$$

  With the above equations, the performance of the system can
be computed. Notice that the procedure itself is independent of
whether or not the system is being jammed. The presence or
absence of a jammer will determine how many channels are availa-
ble and hence what the GOS will be. Therefore there is a direct
coupling between the performance monitoring procedure described
in Chapter 2 and the capacity allocation scheme currently being
discussed. A simple flow chart is shown in Figure 3.1. The
boxes labeled "spread BW by 50%" control the number of channels
that are available. For example, if the only means at the dis-
posal of the system at a given point in time to, say halve
the net information bandwidth (and hence allow this now unused
bandwidth to be consumed by extra spreading of the spectrum) is
to reduce the number of available channels, then these boxes
would initiate a command to reduce the number of available
channels by 50%.

  Notice that both the priority arrangements and the specific
channel assignment can be performed by the same "switch".
All that is necessary is for the switch to keep information
about which priority class users are occupying which channels.
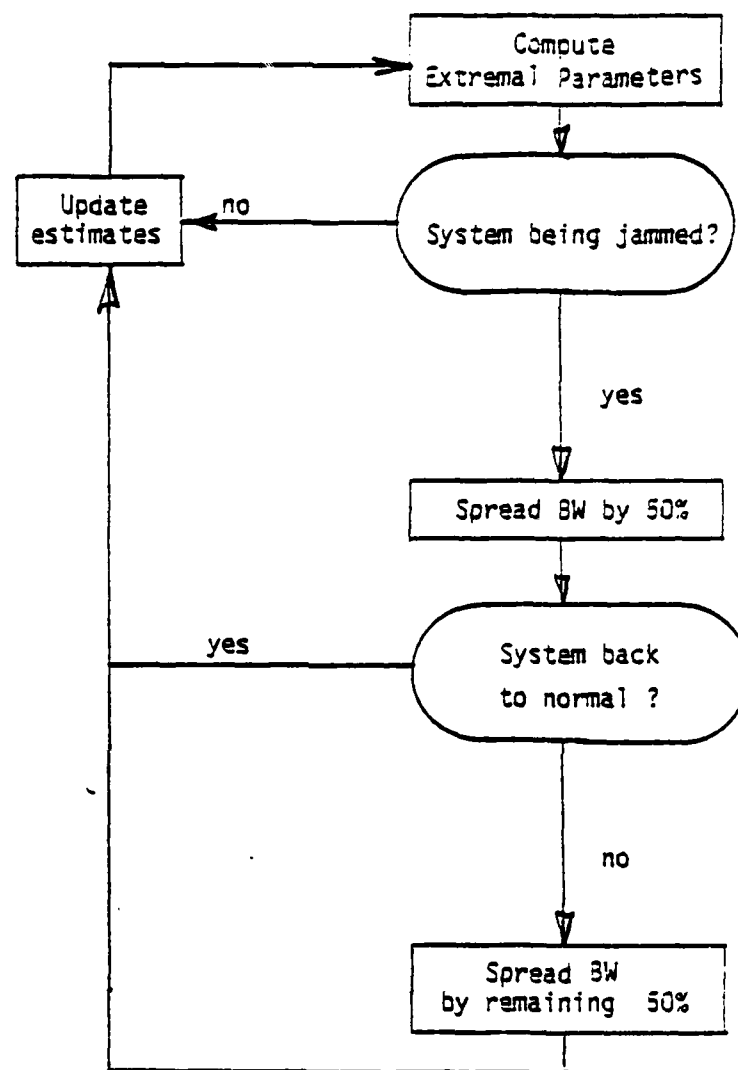
Figure 3.1

Flow Chart for Jam-sensing System Control

In that way the switch will be able to bump a #3 priority class user when necessary (i.e. when all channels are occupied and a #1 priority class user requests a channel) as opposed to possibly bumping a user from either classes #1 or #2 by mistake. Finally, if at a given point in time when it is necessary to invoke this procedure there are $c_3 \leq c_0$ #3 class users occupying channels, the choice of precisely which #3 to bump can be made in a variety of ways (e.g. randomly, the last to access a channel, etc.

Equations (3.5), (3.7), and (3.11) were evaluated numerically for several combinations of traffic parameters and the results are presented in Table 3.3. The assumption implicit in the traffic values chosen in Table 3.3 is that the highest priority users (i.e., users in class #1) only transmit a small amount of traffic whereas the lowest priority users (i.e., class #3) have by far the greatest amount of traffic to transmit.

Upon examining the results of Table 3.3, it can be seen that, at least for the traffic values chosen, the GOS for user class #1 is almost perfect (i.e. less than 10^{-5}) except when there are only three channels and even in that case, it is less than 10^{-3}. On the other hand, the GOS for user class #3 is uniformly poor except when 24 channels are used in combination with $E_1=1$, $E_2=2$, and $E_3=10$ erlangs. Both of these results are direct consequences of the priority assignment (i.e. the ability of #1 user to "bump" a #3 user).

Also, as seen from Eq. (3.7), the GOS of a user in class #2

is just that which would be predicted by using the Erlang B formula in which the total number of channels and the total traffic (i.e. $E = E_1 + E_2 + E_3$) are used. It is seen that if say, E_3 is 10 erlangs and E_2 is 2 erlangs, GOS_2 will seriously degrade if the total number of channels C is reduced from 24 to 12. Therefore, for the particular parameter values used in Table 3.3, this priority scheme only gives true protection in terms of GOS to a user in class #1 when the overall system is being jammed severely enough to cause a significant reduction (e.g. 50%) in the number of channels that are available for use.

However, all the users in all priority classes that gain access to the channels receive the added protection against the jammer that the increased processing gain allows. This can be illustrated very simply by an example. Suppose there are initially 24 channels available and the offered traffic of each of the three user classes is as follows:

$$E_1 = 0.5$$
$$E_2 = 2$$
$$E_3 = 10$$

Assume the system is operating normally (i.e. it is not being jammed) and no priority assignment is necessary. From the Erlang B formula, the GOS of any user will be 1.3×10^{-3}. Note that this applies to any user in any priority class.

If a jamming signal is now detected, one can invoke the priority system. If this is done without changing the total number of channels (i.e. C still equals 24), the protection against the jammer will be whatever it would have been without the priority

- 34 -

Table 3.5

Grade of Service Dependence on Channel Allocation

C	E_1	E_2	E_3	GOS_1	GOS_2	GOS_3 (upper bound)
24	.5	2	10	$.2 \times 10^{-33}$	$.13 \times 10^{-2}$	.33
12	.5	2	10	$.71 \times 10^{-12}$	.22	.48
6	.5	2	10	$.75 \times 10^{-5}$	.57	.71
3	.5	2	10	$.34 \times 10^{-2}$	.78	.85
24	.1	2	10	$<10^{-37}$	$.87 \times 10^{-3}$	$.92 \times 10^{-1}$
12	.1	2	10	$.16 \times 10^{-14}$	.20	.28
6	.1	2	10	$.5 \times 10^{-6}$	.56	.60
3	.1	2	10	$.11 \times 10^{-2}$	.77	.79
24	.5	2	20	$.11 \times 10^{-34}$	.12	.41
12	.5	2	20	$.29 \times 10^{-14}$	.51	.67
6	.5	2	20	$.31 \times 10^{-6}$	.75	.83

assignment. However, the GOS for a user in class #1 will decrease to less than 10^{-33} (i.e. it will effectively have a dedicated channel) while the GOS for a user in class #3 will become greater than 0.33. A class #2 user of course will continue to have a GOS of 1.3×10^{-3}.

If the jamming signal is reasonably strong, this will not be a viable procedure. That is, some additional processing gain will be needed and therefore suppose that the system goes to the priority scheme and in addition reduces the number of channels from 24 to 12. This will yield an extra 3dB in processing gain agains the jammer and for all practical purposes will not effect the GOS of a user in class #1 (it will now be less than 10^{-12}). However, it will substantially degrade the GOS for a user in class #2. Instead of being 1.3×10^{-3}, it will now be 0.22. Also, GOS_3 will increase from 0.33 to 0.48.

The final question then is what has this extra 3 dB in processing gain bought us? The answer to this question will be deferred until Chapter 7 in which the performance of various modulation/spreading techniques under different jamming conditions is discussed.

In light of the above discussion, it can be seen that the combination of the priority system with a variable number of channels which is determined by the condition of the channels (i.e. whether or not they are being jammed) can provide the necessary channel accessibility to users having the highest priority plus increased jammer protection to all users who gain access to a channel. Finally, the specific priority scheme chosen here was

Only meant to be indicative of the type of strategy one might employ, and clearly numerous similar schemes (e.g. a #1 user bumps both #2 and #3 users, while a #2 user bumps #3 users) would be considered.

Conclusions

In summary, it has been demonstrated that a simple scheme to dynamically allocate channel capacity on a priority basis can, when used in conjunction with a procedure to monitor the condition of the channels, guarantee access to the channels to the high priority users, plus guarantee improved performance to _any_ of the users who are able to access a channel. This improved performance arises because the process of reducing the total number of channels in use (while still providing a good GOS to high priority users) allows the remaining channels to spread their bandwidths to the greatest extent possible and hence combat a jamming signal much more successfully then they otherwise would have been able to do.

CHAPTER 4

Susceptibility Analysis of a Frame Synchronization Technique for Multichannel Frequency-Hopped BNCFSK Systems in Partial-Band Noise or CW-Tone Jamming

4.1 Introduction

Jamming can affect a communication system in several ways. It can attack the information-bearing signal directly, or it can reduce system effectiveness through attacks on phase synchronization, spreading-code synchronization, etc. In this chapter we consider the vulnerability to jamming of a frame synchronization technique for a multichannel frequency-hopped binary noncoherent frequency-shift-key system. We investigate the cases of partial-band noise or CW-tone jamming. The derivations of the key results will be presented in as general a form as possible, in order to permit extensive parametric studies and optimizations. The formulas are applicable to many special cases.

We let N_c represent the number of time-channels, with the last one being the frame-sync channel. The receiver looks for the time-channel which exhibits a particular frame-sync pattern of symbols over an interval of several frames. For the first N_c-1 time-channels, because of the randomness of the data within them, there is normally a very small probability that one of them would be mistaken for the sync channel. Jamming may increase this probability, and also may increase the probability that the correct channel may be missed during the search for frame synchronization.

It is of interest to study the system vulnerability for non-spread and also frequency-hopped systems. For the frequency-hopped cases, we let N_s represent the total number of frequency slots available for use, with two selected (for this binary analysis) at any one time as "mark" and "space" frequencies. In a partial-band jamming strategy, the jamming waveform produces energy in K_s of the N_s slots, with K_s allowed to be, conceptually, any number from one to N_s. We assume that the system hops once per time-channel, in a pseudo-random manner which is unknown to the jammer. It is further assumed that the receiver is synchronized to the spreading code.

In the following section we derive certain key general results. These are then used to obtain specific results, which are then presented in the next section.

4.2 Derivation of General Results

For the results obtained here, it is not necessary that we consider specific jamming cases. These follow in the next section. Let us assume that the receiver has no prior knowledge of the correct sync channel, and that it begins a search for a frame-sync pattern in the ith time slot. The frame-sync pattern is one of alternating "ones" and "zeros". The first time-slot searched can therefore be any one of the N_c, with equal probability.

The testing for a particular time-slot involves checking the n_s consecutive binary symbols in that slot over n_s consecutive frames. If at least b of the n_s match the sync

pattern, then "Frame Sync" is declared by the receiver. Otherwise, the testing ends for that slot and is begun for the next slot. This continues until a match is obtained. The results obtained here provide the means for optimal choices of both n_s and b.

The following fundamental probabilities are required:

(a) $P_i(FS)$ = probability of a false "Frame-Sync" decision for the ith time-slot

(b) $P_i(CD)$ = probability of a correct dismissal decision for the ith time-slot

$= 1 - P_i(FS)$

(c) $P_i(CS)$ = probability of a correct "Frame-Sync" decision for the ith time-slot ($i = N_c$)

(d) $P_i(FD)$ = probability of a false dismissal decision for the ith time-slot ($i = N_c$)

$= 1 - P_i(CS)$

We now observe that the conditional probability of a correct "Frame Sync" decision, given that the search starts on the ith time-slot, is

P[correct "Frame Sync" | search starts on the ith slot]

$= [P_i(CD)]^{N_c-i} \cdot [P_i(CS)]$

$+ [P_i(CD)]^{N_c-i} \cdot [P_i(FD)] \cdot [P_i(CD)]^{N_c-1} \cdot [P_i(CS)]$

$+ \ldots$

$$= \sum_{j=0}^{\infty} [P_i(CD)]^{N_c-i} \cdot [P_i(CS)] \cdot [P_i(FD)]^j \cdot [P_i(CD)]^{j(N_c-1)} \qquad (4.1)$$

- 41 -

In (4.1), the jth term is the probability of a correct "Frame Sync" on the $(j + 1)^{th}$ test, given that search started on the ith slot.

Therefore,

P[correct "Frame Sync"]

$$= \sum_{j=0}^{\infty} \sum_{i=1}^{N_c} \frac{1}{N_c} \cdot [P_i(CD)]^{N_c-i} \cdot [P_i(CS)] \cdot [P_i(FD)]^j$$

$$\cdot [P_i(CD)]^{j(N_c-1)} \tag{4.2}$$

But, since

$$\sum_{i=1}^{N_c} X^{N_c-i} = X^{N_c-1} + X^{N_c-2} + \cdots + X + 1$$

$$= \frac{1 - X^{N_c}}{1 - X} \tag{4.3}$$

We obtain

P[correct "Frame Sync"]

$$= \sum_{j=0}^{\infty} \frac{1}{N_c} \cdot \frac{[P_i(CS)] \cdot [P_i(FD]^j \cdot [P_i(CD)]^{j(N_c-1)}}{[P_i(FS)]}$$

$$\cdot \{1 - [P_i(CD)]^{N_c}\} \tag{4.4}$$

Additional simplification results from

$$\sum_{j=0}^{\infty} X^j = 1 + X + X^2 + \cdots$$

$$= \frac{1}{1 - X} \tag{4.5}$$

with

$$X = [P_i(FD)] \cdot [P_i(CD)]^{N_c-1} \qquad (4.6$$

This leads to

P[correct "Frame Sync"]

$$= \frac{[P_i(CS)] \cdot \{1 - [P_i(CD)]^{N_c}\}}{N_c[P_i(FS)] \cdot \{1 - [P_i(FD)] \cdot [P_i(CD)]^{N_c-1}\}} \qquad (4.7$$

A statistic of some interest is E[j], defined and evaluated below.

$$E[j] = \sum_{j=0}^{\infty} \frac{j}{N_c} \cdot \frac{[P_i(CS)] \cdot [P_i(FD)]^j \cdot [P_i(CD)]^{j(N_c-1)}}{[P_i(FS)}$$

$$\cdot \{1 - [P_i(CD)]^{N_c}\} \qquad (4.8$$

Using

$$\sum_{j=0}^{\infty} j \cdot X^j = X + 2X^2 + 3X^3 + \cdots$$

$$= \frac{X}{(1 - X)^2} \qquad (4.9$$

yields

$$E[j] = \frac{[P_i(CS)] \cdot \{1 - [P_i(CD)]^{N_c}\} \cdot [P_i(FD)] \cdot [P_i(CD)]^{N_c-1}}{N_c \cdot [P_i(FS)] \cdot \{1 - [P_i(FD)] \cdot [P_i(CD)]^{N_c-1}\}^2} \qquad (4.10$$

Perhaps of more interest than E[j] is E[j|correct "Frame Sync"] which is found by dividing (4.10) by (4.7).

This gives

$$E[j \mid \text{correct "Frame Sync"}]$$

$$= \frac{[P_i(FD)] \cdot [P_i(CD)]^{N_c-1}}{1 - [P_i(FD)][P_i(CD)]^{N_c-1}} \tag{4.1}$$

In using (4.10) or (4.11), it should be recalled that $(j+1)$ represents the number of attempts before "Frame Sync" is declared.

Another function of interest is the probability that j is equal to or less than some value j_o. This is derived below.

$$P[j \leq j_o]$$

$$= \sum_{j=0}^{j_o} \frac{[P_i(CS)] \cdot [P_i(FD)]^j \cdot [P_i(CD)]^{j(N_c-1)} \cdot \{1 - [P_i(CD)]^{N_c}\}}{N_c \cdot [P_i(FS)]} \tag{4.1}$$

This is simplified by using

$$\sum_{j=0}^{j_o} X^j = 1 + X + \cdots + X^{j_o}$$

$$= \frac{1 - X^{j_o+1}}{1 - X} \tag{4.1}$$

yielding

$$P[j \leq j_o]$$

$$= \frac{[P_i(CS)] \cdot \{1 - [P_i(CD)]^{N_c}\} \cdot \{1 - ([P_i(FD)] \cdot [P_i(CD)]^{N_c-1})^{j_o+1}\}}{N_c \cdot [P_i(FS)] \cdot \{1 - [P_i(FD)] \cdot [P_i(CD)]^{N_c-1}\}} \tag{4.1}$$

- 44 -

Equation (4.14) could be used in solving for the value j_o such that $P[j \leq j_o]$ is a given value, such as 0.9, 0.95, etc. Thus, changes in j_o, as signal and/or jamming conditions change, represents a meaningful criterion for comparison (as do (4.7) and (4.11)).

We next derive the general results, corresponding to the above, for the false "Frame Sync" case. Consider, first, a few special cases for the probability of a false "Frame Sync" decision on the $(\ell+1)^{th}$ attempt, given that search stated on the ith time-slot. For example,

P[False "Frame Sync" on 1st Attempt | Search Starts in i^{th} Slot]

$$= 1 - [P_i(CD)]^{N_c - i} \tag{4.15}$$

P[False "Frame Sync" on 2nd Attempt | Search Starts on i^{th} Slot]

$$= [P_i(CD)]^{N_c - i} \cdot [P_i(FD)] \cdot \{1 - [P_i(CD)]^{N_c - 1}\} \tag{4.16}$$

P[False "Frame Sync" on 3rd Attempt | Search Starts in i^{th} Slot]

$$= [P_i(CD)]^{N_c - i} \cdot [P_i(FD)]^2 \cdot [P_i(DC)]^{N_c - 1} \cdot \{1 - [P_i(DC)]^{N_c - 1}\} \tag{4.17}$$

The general result is

P[False "Frame Sync on $(\ell+1)^{th}$ attempt | Search Starts on i^{th} Slot]

$$= 1 - [P_i(CD)]^{N_c - 1} \qquad\qquad \text{for } \ell = 0$$

$$= [P_i(CD)]^{N_c - 1} \cdot [P_i(FD)]^\ell \cdot [P_i(CD)]^{(\ell-1)(N_c-1)} \{1 - [P_i(CD)]^{N_c - 1}\},$$

$$- 45 - \qquad \text{for } \ell = 1,2,3,\ldots \tag{4.18}$$

Thus we have

P[False "Frame Sync" on 1st Attempt]

$$= \sum_{i=1}^{N_C} \frac{1}{N_C} \cdot \{1-[P_i(CD)]^{N_C-i}\}$$

$$= 1 - \frac{\{1-[P_i(CD)]^{N_C}\}}{N_C \cdot \{1-[P_i(CD)]\}}$$

$$= 1 - \frac{\{1-[P_i(CD)]^{N_C}\}}{N_C \cdot [P_i(FS)]} \tag{4.19}$$

Equation (4.19) follows from the result

$$\sum_{i=1}^{N_C} X^{N_C-i} = X^{N_C-1} + X^{N_C-2} + \cdots + X + 1$$

$$= \frac{1-X^{N_C}}{1-X} \tag{4.20}$$

For the other cases, $(\imath=1,2,3,\cdots)$, we have

P[False "Frame Sync" on $(\imath+1)^{th}$ test]

$$= \sum_{i=1}^{N_C} \frac{1}{N_C} \cdot [P_i(CD)]^{N_C-i} \cdot [P_i(FD)]^{\imath} \cdot [P_i(CD)]^{(\imath-1)(N_C-1)}$$

$$\cdot \{1-[P_i(CD)]^{N_C-1}\}$$

$$= \frac{[P_i(FD)]^{\imath} \cdot [P_i(CD)]^{(\imath-1)(N_C-1)} \cdot \{1-[P_i(CD)]^{N_C-1}\}\{1-[P_i(CD)]^{N_C}\}}{N_C \cdot \{1-[P_i(CD)]\}}$$

- 46 -

$$= \frac{\{[P_i(FD)] \cdot [P_i(CD)]^{N_c-1}\}^{\ell} \cdot \{1-[P_i(CD)]^{N_c-1}\} \cdot \{1-[P_i(CD)]^{N_c}\}}{N_c \cdot [P_i(CD)]^{N_c-1} \cdot [P_i(FS)]} \qquad (4.20)$$

Now we evaluate

P[False "Frame Sync"]

$$= \sum_{\ell=0}^{\infty} P[\text{False "Frame Sync" on } (\ell+1)^{th} \text{ test}]$$

$$= \frac{(N_c-1)-N_c[P_i(CD)]+[P_i(CD)]^{N_c}}{N_c \cdot [P_i(FS)]}$$

$$= \frac{[P_i(FD)] \cdot \{1-P_i(CD)]^{N_c-1}\} \cdot \{1-[P_i(CD)]^{N_c}\}}{N_c \cdot [P_i(FS)] \cdot \{1-[P_i(FD)] \cdot [P_i(CD)]^{N_c-1}\}} \qquad (4.21)$$

where the first term of (4.21) is (4.19) in different form, and
the second term uses (4.20) for the summation over the ($\ell \neq 0$)
values.

In a similar manner for defining E[j], equation (4.8),
we now define and evaluate E[ℓ] for this false "Frame Sync"
case.

$$E[\ell] = \sum_{\ell=0}^{\infty} \ell \cdot P[\text{False "Frame Sync" on } (\ell+1)^{th} \text{ test}] \qquad (4.22)$$

$$= \frac{\{1-[P_i(CD)]^{N_c-1}\} \cdot \{1-[P_i(CD)]^{N_c}\}}{N_c[P_i(CD)]^{N_c-1} \cdot [P_i(FS)]} \cdot \sum_{\ell=1}^{\infty} \ell \cdot \{[P_i(FD)] \cdot [P_i(CD)]^{N_c-1}\}^{\ell}$$

Again using (4.9), with the $\ell=0$ term dropping out, we obtain

$$E[\ell] = \frac{\{1-[P_i(CD)]^{N_c-1}\}\cdot\{1-[P_i(CD)]^{N_c}\}\cdot[P_i(FD)]}{N_c\cdot[P_i(FS)]\cdot\{1-[P_i(FD)]\cdot[P_i(CD)]^{N_c-1}\}^2} \tag{4.23}$$

To find $E[\ell|\text{False "Frame Sync"}]$, divide (4.23) by (4.21).

Now we find the probability, for this false "Frame Sync" case, which is analogous to (4.14).

$$P[\ell \leq \ell_0] = \sum_{\ell=0}^{\ell_0} P[\text{False "Frame Sync" on } (\ell+1)^{th}\text{test}]$$

$$= \frac{(N_c-1)-N_c\cdot[P_i(CD)]+[P_i(CD)]^{N_c}}{N_c\cdot[P_i(FS)]} \tag{4.24}$$

$$+ \frac{\{1-[P_i(CD)]^{N_c-1}\}\cdot\{1-[P_i(CD)]^{N_c}\}}{N_c[P_i(CD)]^{N_c-1}\cdot[P_i(FS)]} \sum_{\ell=1}^{\ell_0}\{[P_i(FD)]\cdot[P_i(CD)]^{N_c-1}\}^{\ell}$$

To simplify (4.24), we can use

$$\sum_{\ell=1}^{\ell_0} X^{\ell} = X + X^2 + \cdots + X^{\ell_0} \tag{4.25}$$

$$= \frac{X(1-X^{\ell_0})}{1-X}$$

The result is

$$P[\ell \leq \ell_0] = \frac{(N_c-1) \cdot N_c [P_i(CD)] + [P_i(CD)]^{N_c}}{N_c \cdot [P_i(FS)]}$$

$$+ \frac{\{1-[P_i(CD)]^{N_c-1}\} \cdot \{1-[P_i(CD)]^{N_c}\} \cdot [P_i(FD)] \cdot [P_i(CD)]^{N_c-1})^{\ell_0}]}{N_c \cdot [P_i(FS)] \cdot \{1-[P_i(FD)] \cdot [P_i(CD)]^{N_c-1}\}}$$

(4.2?)

The key equations above represent the desired set of gen-
eral results, which provide the basis for evaluation of the
frame synchronization technique with specific jamming waveforms.
In the next section we consider the application of the above
equations to special cases of interest.

4.3 General Results Applied to Specific Jamming Cases

The general results presented in the previous section
provide the framework for study of the effects of jamming on
the frame-synchronization technique described. We begin by
noting that, for most jamming cases of interest for the NCFSK
system being considered, the channel is a binary symmetric
channel. That is, the two types of errors occur with equal
probability, which we represent by the symbol p.

For the first N_c-1 time-slots, the receiver will produce
"zero" and "one" at its output with equal probability. This
is due, of course, to the completely random nature of the data
in these slots and the symmetric nature of the channel. As
a consequence, we have

$$P_i(CD) = \sum_{j=n_s-b+1}^{b-1} \binom{n_s}{j} \left(\frac{1}{2}\right)^j \left(\frac{1}{2}\right)^{n_s-j}$$

$$= \frac{1}{2^{n_s}} \cdot \sum_{j=n_s-b+1}^{b-1} \binom{n_s}{j}$$

and

$$P_i(FS) = 1 - P_i(CD)$$

$$= \sum_{j=0}^{n_s-b} \binom{n_s}{j} \left(\frac{1}{2}\right)^j \left(\frac{1}{2}\right)^{N_s-j} + \sum_{j=b}^{n_s} \binom{n_s}{j} \left(\frac{1}{2}\right)^j \left(\frac{1}{2}\right)^{N_s-j}$$

$$= \frac{2}{2^{n_s}} \cdot \sum_{j=0}^{n_s-b} \binom{n_s}{j}$$

Thus $P_i(CD)$ and $P_i(FS)$, for substitution into the key equations of the previous section, are independent of jamming, since they are not functions of the bit error probability p. Instead, they are functions only of n_s, the number of symbols tested for the sync pattern, and b, the threshold for determining if "Frame Sync" is to be declared.

On the other hand, $P_i(CS)$ and $P_i(FD)$, which apply only for time-slot N_c where the non-random sync pattern is located, are functions of p, and therefore depend on the jamming waveform assumed. These are found from

$$P_i(CS) = \sum_{j=0}^{n_s-b} \binom{n_s}{j} (1-p)^j (p)^{n_s-j} + \sum_{j=b}^{n_s} \binom{n_s}{j} (1-p)^j (p)^{n_s-j}$$

and

$$P_i(FD) = 1 - P_i(CS)$$

$$= \sum_{j=n_s-b+1}^{b-1} \binom{n_s}{j}(1-p)^j(p)^{n_s-j}$$

(4.30)

The remaining task is to list, for various jamming wave-forms, the appropriate expressions for p. They are presented below.

(1) <u>Random Noise Only, Without Jamming, No FH</u>

$$p = P_{ao}$$

$$= \frac{1}{2}\exp[-\frac{S}{N}]$$

(4.31)

(2) <u>Random Noise, with one Frequency Slot Subjected</u>
<u>to Noise Jamming, No FH</u>

$$p = P_{a1}$$

$$= \frac{1}{2}\exp\left[-\frac{(\frac{S}{N})(\frac{S}{J_s})}{(\frac{S}{N})+2(\frac{S}{J_s})}\right]$$

(4.32)

(3) <u>Random Noise, With Both Frequency Slots Subjected</u>
<u>to Noise Jamming, No FH</u>

$$p = P_{a2}$$

$$= \frac{1}{2}\exp\left[-\frac{(\frac{S}{N})(\frac{S}{J_s})}{2(\frac{S}{N})+2(\frac{S}{J_s})}\right]$$

(4.33)

- 51 -

(4) <u>Random Noise, with One Frequency Slot Subjected</u>
<u>to CW-Tone Jamming, No FH</u>

$$p = P_{b1}$$

$$= \frac{1}{2} \cdot Q\left[\frac{(\frac{S}{N})^{1/2}}{(\frac{S}{J_s})^{1/2}} \, , \, (\frac{S}{N})^{1/2}\right] \qquad (4.3\cdot$$

(5) <u>Random Noise, with Both Frequency Slots Subjected</u>
<u>to CW-Tone Jamming, No FH</u>

$$p = P_{b2}$$

$$= \frac{1}{2} \qquad\qquad (4.35)$$

$$= \frac{1}{2\pi} \int_0^\pi \left[Q\left(\left[\frac{S}{N} + \frac{2(\frac{S}{N})\cos\theta}{(\frac{S}{J_s})^{1/2}} + \frac{(\frac{S}{N})^{1/2}}{(\frac{S}{J_s})^{1/2}}\right], \left[\frac{(\frac{S}{N})^{1/2}}{(\frac{S}{J_s})^{1/2}}\right]\right) \right.$$

$$\left. - Q\left(\left[\frac{(\frac{S}{N})^{1/2}}{(\frac{S}{J_s})^{1/2}}\right] \, , \, \left[(\frac{S}{N}) + \frac{2(\frac{S}{N})\cos\theta}{(\frac{S}{J_s})^{1/2}} + \frac{(\frac{S}{N})^{1/2}}{(\frac{S}{J_s})^{1/2}}\right]\right) \right] \cdot d\theta$$

(6) <u>Random Noise, With Noise Jamming in K_s of N_s</u>
<u>Frequency Slots, Frequency Hopping</u>

$$p = P_{ak}$$

$$= (\frac{K_s}{N_s})(\frac{K_s-1}{N_s-1}) \cdot P_{a2} + (\frac{2K_s}{N_s})(\frac{N_s-K_s}{N_s-1}) \cdot P_{a1} \qquad (4.36$$

$$+ (\frac{N_s-K_s}{N_s})(\frac{N_s-K_s-1}{N_s-1}) \cdot P_{ao}$$

(In (4.36), use (4.31), (4.32), and (4.33).)

(7) Random Noise, with CW-Tone Jamming in K_s of N_s

Frequency Slots, Frequency-Hopping

$$p = P_{bK}$$

$$= (\frac{K_s}{N_s})(\frac{K_s-1}{N_s-1}) \cdot P_{b2} + (\frac{2K_s}{N_s})(\frac{N_s-K_s}{N_s-1}) \cdot P_{b1} \qquad (4.37$$

$$+ (\frac{N_s-K_s}{N_s})(\frac{N_s-K_s-1}{N_s-1}) \cdot P_{ao}$$

(In (4.37), use (4.31), (4.34), and (4.35).)

For the above equations, (S/N) and (S/J_s) represent
average power signal-to-noise and signal-to-jamming ratios
at the detector input for the branch which contains the signal
component. In addition, Q(x,y) is the Q-Function defined by

$$Q(x,y) = \int_y^\infty t \cdot exp \left[-(\frac{t^2+x^2}{2})\right] \cdot I_o(xt) \cdot dt \qquad (4.38$$

For the frequency-hopping cases, equations (4.36) and (4.37),
the following substitution is used.

$$J_s = \frac{J}{K_s} \qquad (4.\text{—})$$

with J then representing the total jamming power over the spread bandwidth.

In addition to the above seven jamming cases, there are two others for which we present results. Each of these represents an example for the binary nonsymmetric channel which requires two error-probabilities to characterize it. If we let P_o be probability-of-error for a "zero" transmitted and p_1 be the probability-of-error for a "one" transmitted, it follows directly that

P["zero" received]

$$= P(0_r) = \frac{1 - P_o P_i}{2} \qquad (4.4\text{—})$$

and

P["one" received]

$$= P(1_r) = \frac{1 + P_o - P_1}{2} \qquad (4.4\text{—})$$

We must find the new expressions, for these two cases, for $P_i(FS)$, $P_i(CS)$, etc. The results below apply to the case n_s even and b at least as large as one-half of n_s. Similar results can be found for n_s odd. After considerable effort we obtain

$$P_i(FS) = 1 - P_i(CD)$$

$$= \sum_{j=0}^{n_s-b} \sum_{i=0}^{j} 2 \cdot [P(0_r)]^{\frac{n_s-j+2i}{2}} \cdot [P(1_r)]^{\frac{n_s+j-2i}{2}} \cdot \binom{\frac{n_s}{2}}{j-i}\binom{\frac{n_s}{2}}{i} \quad (4.4$$

and

$$P_i(CS) = 1 - P_i(CD)$$

$$= \sum_{j=0}^{n_s-b} \sum_{i=0}^{j} \binom{\frac{n_s}{2}}{j-i} \cdot \binom{\frac{n_s}{2}}{i} \cdot \left[(1-p)^{\frac{n_s}{2}-j+1} \cdot p_1^{j-i} \cdot (1-p_0)^{\frac{n_s}{2}i} \cdot p_0^i \right.$$

$$\left. + p_1^{\frac{n_s}{2}-j+i} \cdot (1-p_1)^{j-i} \cdot p_0^{\frac{n_s}{2}-i} (1-P_0)^i \right] \quad (4.4$$

As with the first seven cases, only the binary-symbol error-probability remains to be specified in order to define a particular case. The two of interest are specified below.

(3) **Random Noise, With One Fixed Frequency Slot Subjected to Noise Jamming, No FH**

$$P_0 = \frac{N}{2N+J_s} \cdot \exp\left[-\frac{S}{2N+J_s}\right]$$

(4.44)

$$= \frac{1}{2+\frac{(\frac{S}{N})}{(\frac{S}{J_s})}} \cdot \exp\left[-\frac{(\frac{S}{N})(\frac{S}{J_s})}{2(\frac{S}{J_s})+(\frac{S}{N})}\right]$$

$$P_1 = \frac{N+J_s}{2N+J_s} \cdot \exp\left[\frac{S}{2N+J_s}\right]$$

$$\frac{(\frac{S}{N})+(\frac{S}{J_s})}{2(\frac{S}{J_s})+(\frac{S}{N})} \cdot \exp-\left[\frac{(\frac{S}{N})(\frac{S}{J_s})}{2(\frac{S}{J_s})+(\frac{S}{N})}\right]$$

(4.45)

(9) **Random Noise, With One Fixed Frequency Slot Subjected to CW-Tone Jamming, No FH**

$$P_0 = \frac{1}{2}\exp\left[-\frac{1}{2}\{(\frac{S}{N})+\frac{(\frac{S}{N})}{(\frac{S}{J_s})}\}\right]$$

(4.46)

$$I_0\left[\frac{(\frac{S}{N})}{(\frac{S}{J_s})^{1/2}}\right]$$

with $I_o(x)$ being the modified Bessel funciton of the first kind,

$$P_I = \frac{1}{2}\left[1-Q \left\{ \left(\tfrac{S}{N}\right)^{1/2}, \frac{\left(\tfrac{S}{N}\right)^{1/2}}{\left(\tfrac{S}{J_s}\right)^{1/2}} \right\} \right.$$

$$\left. +Q \left\{ \frac{\left(\tfrac{S}{N}\right)^{1/2}}{\left(\tfrac{S}{J_s}\right)^{1/2}}, \left(\tfrac{S}{N}\right)^{1/2} \right\} \right]$$

(4.?

In the next section we summarize the above results and discuss the steps involved in using them.

4.4 Discussion of Results

The above set of comprehensive results provides the analytical tools for extensive studies of the effects of jamming on frame synchronization for the multichannel tactical system. Both spread and unspread cases are included, along with a variety of jamming waveforms and strategies. These same tools can be applied toward system or jamming optimization, requiring probably many sets of performance curves in order to locate "optimum" values, etc.

Since there are many cases included in the above, it seems highly desirable to provide the following table which summarizes the nine cases and gives the equation numbers necessary to apply the formulas.

Summary of Results

I. General equations applicable to all cases

  .4.7, 4.10, 4.11, 4.14, 4.21, 4.23, 4.26

II. Equations for cases 1-7

  4.27, 4.28, 4.29, 4.30

III. Equations for cases 8,9

  4.40, 4.41, 4.42, 4.43

IV. Equations for individual cases

  Case 1 - 4.31

  Case 2 - 4.32

  Case 3 - 4.33

  Case 4 - 4.34

  Case 5 - 4.35

  Case 6 - 4.31, 4.32, 4.33, 4.36

  Case 7 - 4.31, 4.34, 4.35, 4.37

  Case 8 - 4.44, 4.45

  Case 9 - 4.46, 4.47

Even though it would be possible, within the time con-
straints imposed for this task, to present a few results in
graphical form, it is believed that to do so would be mis-
leading to the reader. To reach meaningful conclusions con-
cerning the effects of parameter variations and changes in
ECM/ECCM strategies will require that curves be found for
a large number of potential situations. Only then can design
and/or operating decisions be made with a high degree of con-
fidence. For example, results obtained by the use of the

analytical tools developed here may lead to novel techniques
involving adaptivity of the number of time channels, frame-
sync pattern, pattern length, threshold setting, jamming
waveform, jamming strategy, spreading technique, etc. It is
highly recommended that these evaluations be carried out in
an extensive manner during a follow-up effort, with final
graphical results and conclusions presented in handbook
fashion for ease of application.

However, in order to illustrate the application of the
above tools, the following sample numerical results are listed.

Parameter values selected

n_s = 6
b = 6
N_c = 12
S/N = 16dB

Example 1 Noise Jamming in both frequency slots, no fre-
quency-hopping, S/J = 11.5dB
Result: P(correct "Frame Sync") = 0.844

Example 2 Noise Jamming, frequency hopping with N_s=1000,
Partial-band jamming, P (correct "Frame Sync") =
0.844.

Results: | K_s | S/J |
|---|---|
| 1 | 4.5 dB |
| 2 | 3 dB |
| 100 | -9 dB |

Example 5   Noise Jamming in both frequency slots, no fre-
quency-hopping, S/J = 2dB

Result: P (correct "Frame Sync") = 0.671

CHAPTER 5

Low Rate Error Correcting Codes
In Spread Spectrum Systems

Section 1.   Introduction

Direct sequence spectrum spreading techniques for use over low signal-to-noise ratio channels are in common use and well understood. We briefly summarize the salient features.

A binary digital data sequence producing digits at the rate of R_m digits per second is added modulo-two to a predetermined pseudorandom (PN) sequence of binary digits occurring at a rate of $R_c >> R_m$ digits per second. The resulting digit stream is then used to phase modulate an RF carrier and this signal is transmitted over the given channel. At the receiver, and after synchronization, the in-phase modulo-two addition of a replica of the PN sequence to the received demodulated waveform will remove the PN sequence, resulting in a data sequence approximating that of the transmitter.

The quality of this approximation, as expressed by the bit error rate in the receiver output, is normally a function of the so called processing gain, defined by

$$P.G. = \frac{R_c}{R_m} \quad ,$$

and the signal-to-noise ratio in the channel. In fact, in an additive white Gaussian noise channel of symbol energy to noise power spectral density E_s/N_o, a binary coherent PSK system will

yield a bit error rate equal to that of an unspread channel
with a signal to noise ratio given by

$$\frac{E_b}{N_o} = \frac{E_s}{N_o} \times P.G. \tag{5.1}$$

Although in principle any sequence of binary digits will serve
as a spectrum spreading code, the most commonly used configura-
tion, at least in systems without stringent security requirements,
is the so called maximal length or pseudo-random sequence, whose
basic properties we describe next.

For any positive integer k>1, the maximal length sequence
consists of $2^k - 1$ binary digits, of which 2^{k-1} are ones and $2^{k-1} - 1$
are zeros. Every possible binary k-tuple can be found once
and only once in the sequence. The autocorrelation function
$R(\tau)$ has the value $2^k - 1$ for $\tau = 0$ and -1 for all other integer
values of τ. From the communications point of view these prop-
erties assure us of a low DC component in the binary waveform,
a "random" appearance of the digits and optimal synchronization
performance at the receiver.

Beyond these certainly desirable features, however, the
maximal length sequence has little to recommend itself. In
particular, since the pattern of zeros and ones is always the
same and hence independent of the data sequence, no "coding
gain" beyond the bandwidth--energy--performance tradeoff implicit
in (5-1) is achieved.

The question therefore arises, whether it is possible to
influence that tradeoff by a more intelligent choice of the

sequence of digits transmitted over the channel. In Section 2
to follow we examine this question from the standpoint of error
control coding. In particular, we show that a slight change
in the way maximal length sequences are normally used allows
us to exploit their characteristics as error correcting block
codes. We then derive their basic properties, stipulate an
error correction algorithm and obtain performance values for
the codes in terms of bit error rates and data deletion rates
over the hard quantized Gaussian Channel.

We conclude with the derivation of several asymptotic
bounds on performance and a comparison of bit error rates achieved
under an optimum correlation decoding scheme.

In Section 3 we turn our attention to the idea of using
low rate convolutional codes as a means of spectrum spreading.
We first define a class of codes with good structural properties
and derive their free distance and minimum distance. We then
show that d_{free} for these codes is optimum in the limit as the
constraint length becomes large and compare it with upper bounds
on d_{free} for finite constraint lengths.

Next we give generator and parity check matrices and obtain
an expression for the generator function of the encoder state
transitions. We then use these results to derive upper bounds
on the residual bit error rates when these codes are used on
the binary symmetric and the additive white Gaussian channels
and decoded via the maximum likelihood algorithm.

Finally, we compare the results with more conventional
direct sequence spread spectrum schemes.

Section 2. Maximal Length Sequence Codes

2.1 Code Definitions and Properties

The generation of the pseudo-random sequences discussed in
the introduction is usually accomplished by means of k-stage
shift register, in which the contents of certain stages are
added modulo-two and applied as an input to the first stage.
When such a register is loaded with a particular nonzero combi-
nation of k binary digits and shifted 2^k-1 times, the sequence
corresponding to the starting binary k-tuple. Loading the regis-
ter with any other combination of the digits simply results in
a cyclically shifted output sequence. There are, of course,
exactly 2^k-1 cyclic shifts of a sequence, corresponding to the
2^k-1 nonzero starting k-tuples. If we include as a possiblity
the sequence of 2^k-1 zeros, we have a correspondence between
2^k distinct sequences of binary digits and all 2^k binary k-tuples.
Without great difficulty we may show that these sequences form
a cyclic binary linear (n,k) error correcting code, where $n=2^k-1$.
The weight spectrum of the code is also easily determined and
consists of 2^k-1 sequences of weight 2^{k-1} and one sequence of
weight zero. Finally, the distance between any two code words
is $d=2^{k-1}$ and this implies that the code can correct up to
$e=2^{k-2}-1$ errors.

Invoking Plotkin's bound (1) to the effect that the mini-
mum distance of any linear binary (n,k) block code is bounded
by

$$d \leq \frac{n2^{k-1}}{2^k - 1}$$

We easily show that these codes meet the Plotkin bound with equality and are therefore optimum.

In Figure 5.1 we show the general encoding circuit for these codes and Table 1 lists the corresponding connection pattern, for $3 \leq k \leq 10$.

Since the codes are linear and cyclic, they may be described by a parity check matrix H or a parity check polynomial $h(x)$. The latter is also defined in Table 5.1 and the former is shown in Figure 5.2. Note the convention of forming a codeword with the information digits, denoted by $x_1, x_2, \ldots x_k$, followed by the parity check digits, denoted by $r_1, r_2 \ldots r_{n-k}$, in the usual way for cyclic codes.

As an example, consider the case k=3. In Figure 5.3 we show the encoding circuit and Table 5.2 lists the resulting codewords. In each codeword the first three digits are the information digits and the last four are the parity checks.

Consider now the use of the codes described above as signalling elements in a spectrum spreading application. Since with the exception of the all zero sequence the codewords are maximal length sequences, the balance of zeros and ones and the "randomness" of the digits are roughly preserved, if we assess these properties over sequences much longer than $2^k - 1$. The auto-correlation function is no longer important since every cyclic shift of a sequence is also a legitimate sequence. This fact of course, implies the need for some other means of synchroni-
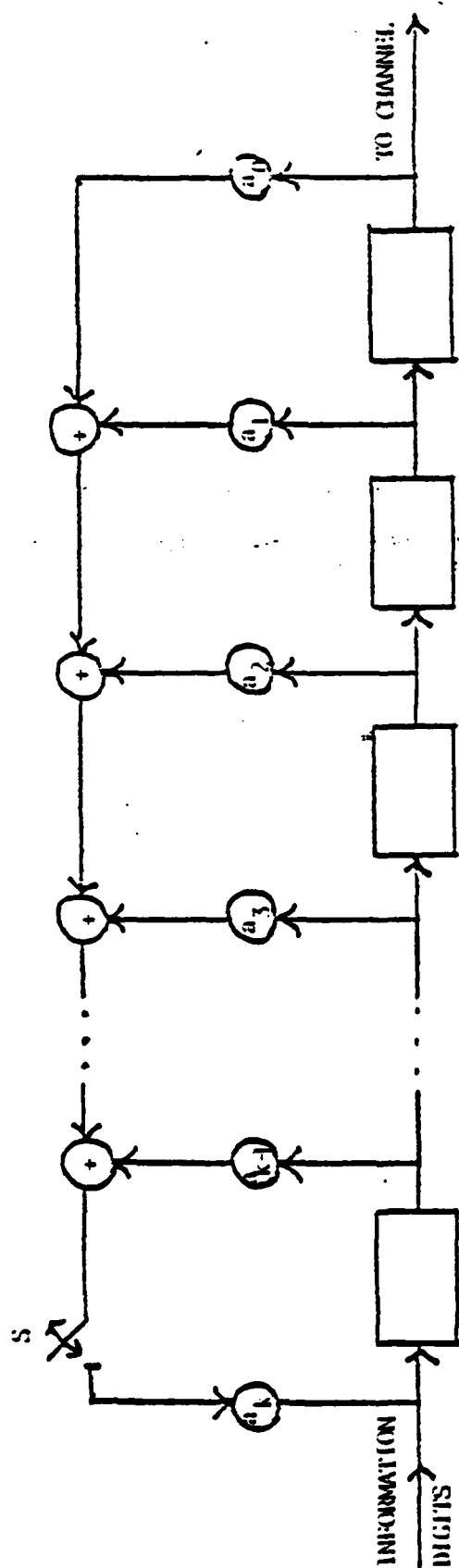
Note: Switch S is open for the k information digits $X_1, X_2, \ldots X_k$ and closed for succeeding n-k parity digits $r_1, r_2, \ldots r_{n-k}$

$$V = X_1 X_2 X_3 \ldots X_k r_1 r_2 \ldots r_{n-k}$$

$$v H^T = 0$$

Figure 5-1 Encoding Circuit for Maximal Length Sequence Codes

- 66 -

$$
H = \begin{bmatrix}
0 & 0 & . & . & . & . & 0 & a_0 & a_1 & a_2 & . & . & . & . & a_{k-1} & a_k \\
. & . & & & . & & & . & . & . & & & & & . \\
. & . & & . & & & & . & . & . & & & & & . \\
. & . & . & & & & & . & . & . & & & & & . \\
. & . & & & & & & . & . & . & & & & & . \\
0 & a_0 & a_1 & . & . & . & & a_{k-2}a_{k-1}a_k & 0. & . & . & . & . & & & 0 \\
a_0 & a_1 & a_2 & . & . & . & & a_{k-1}a_k & 0 & 0 & . & . & . & . & & 0
\end{bmatrix}
$$

Figure 5-2 Parity Check Matrix for Maximal Length Sequence Codes
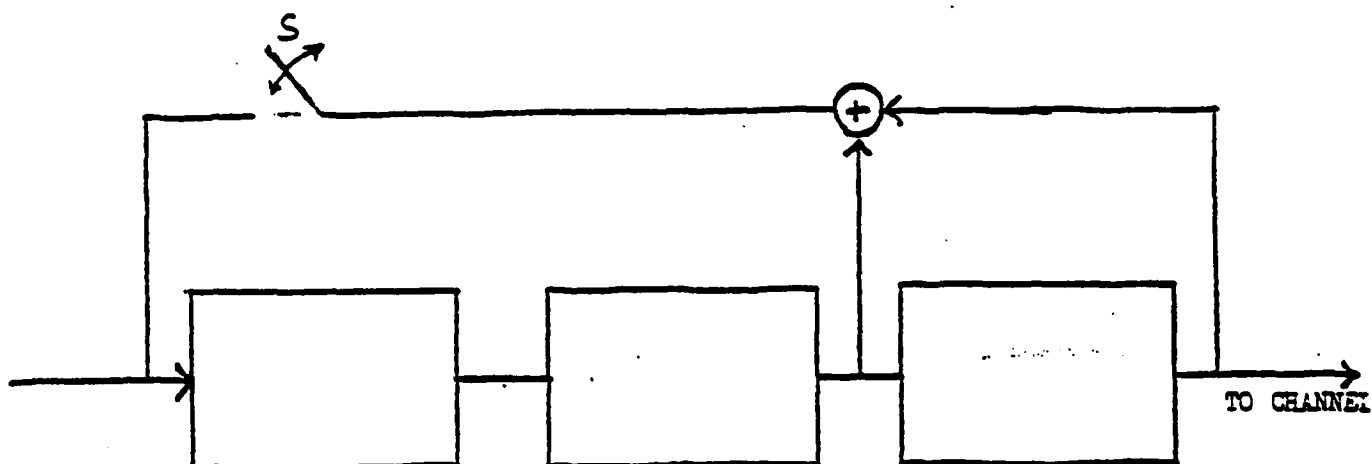
Figure 5-3 Encoding Circuit for k=3 Maximal Length Sequence Code

$h(x) = a_0 + a_1 x + a_2 x^2 \ldots + a_k x^k$		$a_0 a_1 a_2 \ldots a_k$
$k=3$	$1+x+x^3$	1101
$k=4$	$1+x+x^4$	11001
$k=5$	$1+x^2+x^5$	101001
$k=6$	$1+x^2+x^6$	1100001
$k=7$	$1+x^3+x^7$	10010001
$k=8$	$1+x^2+x^3+x^4+x^8$	101110001
$k=9$	$1+x^4+x^9$	1000100001
$k=10$	$1+x^3+x^{10}$	10010000001

Table 5.1

Encoder Connection Patterns and Parity Check
Polynomials for Maximal Length Sequence Codes


```
0 0 0 | 0 0 0 0

0 0 1 | 1 1 0 1

0 1 0 | 0 1 1 1

0 1 1 | 1 0 1 0

1 0 0 | 1 1 1 0

1 0 1 | 0 0 1 1

1 1 0 | 1 0 0 1

1 1 1 | 0 1 0 0
```

Table 5.2

Codewords of (7,3) Maximal Length Sequence Code

zation of the sequences such as a preamble or a cover sequence. The all zero sequence could, if so desired, be eliminated from the code by inserting a one in the data stream after k-1 consecutive zeros prior to encoding and removing it again at the receiver after decoding. This would require some elastic buffering, result in a decrease in data rate and give rise to the possiblity of error propagation in the decoded output.

Aside from these difficulties, however, the codes achieve the effect of spreading the energy in the data sequence, the bandwidth expansion factor being n/k.

2.2 A Hard Decision Decoding Algorithm

We proceed to study the performance of the maximal length sequence codes defined in Section 2.1 under a specific decoding algorithm which, although conceptually quite simple, may entail some significant increase in complexity at the receiver, compared to the standard direct sequence spread spectrum demodulators.

The decoding algorithm is defined as follows:

Let u be a transmitted codeword and v be the corresponding received sequence, differing from u in i places. If the syndrome is zero, accept v. If the syndrome is not zero, change the digits of v successively one at a time, two at a time, and so on up to $e = 2^{k-2} - 1$ at a time, until the resulting sequence has zero syndrome, i.e., is a codeword. Accept this codeword as the correct decision. If none of the sequences have zero syndrome, declare an incorrectable error present. It is clear

- 69 -

that this algorithm works correctly if the number of errors made by the channel noise does not exceed $e = 2^{k-2} - 1$. It is also obvious that an uncorrectable and undetectable error pattern will result in a decoded output error of multiplicity 2^{k-1}, since the decoder in this situation produces an invalid nonzero codeword which must differ from the correct one in 2^{k-1} places.

Finally, if the error pattern is detectable but not correctable, the decoder will pass the received sequence unchanged and therefore the multiplicity of the error pattern at the decoder output will be that of the decoder input.

Thus, if i is the multiplicity of the decoder input error pattern, the decoder output error pattern will be of multiplicity $0, d = 2^{k-1}$ or i. It remains to count the number of decoder input error patterns giving rise to a decoder output error pattern multiplicity in each of the three categories listed above.

Let these quantities be denoted by N_o, N_d and N_i, with the convention that $N_i = 0$ for $i = 0$. N_o, N_d and N_i are, of course, also the number of correctable, uncorrectable and undetectable, and uncorrectable but detectable decoder input error patterns, respectively.

For $i \le e$ we have $N_o = \binom{n}{i}$; $N_d = N_i = 0$, since all input error patterns in this case will be corrected.

For $i = e + 1$, the decoder is unable to produce a codeword since it is restricted to making at most e changes. Thus all input error patterns of multiplicity $e + 1$ are detectable and we have $N_o = N_d = 0$, $N_i = \binom{n}{i}$.

Next, we look at the case $e + 2 \le i \le e + 2 = d$. Since i is beyond

the correction capability of the code, $N_o = 0$. To compute N_d we reason as follows:

The decoder must produce a nonzero codeword by making e or fewer changes. Equivalently, a nonzero codeword must result in a given error pattern of weight i by making e or fewer changes. Now a codeword has exactly d ones and d-1 zeros. If we change j zeros into ones, we must also change d+j-i ones into zeros in order to obtain a vector of weight i. There are $\binom{d-1}{j}$ ways of changing j zeros into ones and for each of these, there are $\binom{d}{d+j-i}$ ways of changing d+j-i ones into zeros. Finally, since the total number of changes, given by j+d+j-i, is limited to e, j may vary from 0 to $[\frac{i-e-2}{2}]$, where the symbol [x] denotes the largest integer not larger than x. In summary, a given nonzero codeword is associated with

$$\sum_{j=0}^{[\frac{i-e-2}{2}]} \binom{d-1}{j} \binom{d}{d+j-i}$$

différent error patterns of weight i and since there are n nonzero codewords we have

$$N_d = n \sum_{j=0}^{[\frac{i-e-2}{2}]} \binom{d-1}{j} \binom{d}{d+j-i}$$

All remaing error patterns of weight i are, of course, detectable by virtue of the fact that they cannot be turned into codewords or with e or fewer changes.

Thus,

$$N_i = \binom{n}{i} - N_d$$

In similar fashion, we derive the values of N_o, N_d, and N_i for $2e+3 \leq 3e+2$ and $3e+3 \leq i \leq 4e+3$. These results are summarized in Table 5.3.

2.3 Performance Over the Binary Symmetric Channel

Consider now the performance of these codes over the binary symmetric channel with error probability p and under the decoding algorithm given above. We evaluate two parameters of code performance.

1. The decoder output bit error rate, defined as
   $$BER = \frac{1}{n} E \text{ \{number of errors in decoder output sequence\}}$$

2. The rate of undetected decoder output bit errors, defined as
   $$UBER = \frac{1}{4} E \text{ \{number of undetected errors in decoder output sequence\}}$$
   Here $E\{x\}$ is the expected value of the random variable x.

   For the situation where decoder output sequences known to be in error are deleted we also define

3. The rate of deleted data bits, given by
   $$DBDR = \frac{k}{n} Pr \text{ \{decoder output sequences is deleted\}}$$
   With the aid of Table 5.3 we then obtain

$$BER = \frac{1}{N} (e+1) \binom{n}{e+1} p^{e+1} q^{n-(e+1)}$$

$$+ \sum_{i=e+2}^{2e+2} dn \sum_{j=0}^{[\frac{i-e-2}{2}]} \binom{d-1}{j} \binom{d}{d+j-i} p^i q^{n-1}$$

- 72 -

	N_o	N_d	N_i
$i < e$	$\binom{n}{i}$	0	0
$i = e+1$	0	0	$\binom{n}{i}$
$e+2 \leq i \leq 2e+2$	0	$n \sum_{j=0}^{[\frac{i-e-2}{2}]} \binom{d-1}{j} \binom{d}{d+j-i}$	$\binom{n}{i} - n \sum_{j=0}^{[\frac{i-e-2}{2}]} \binom{d-1}{j} \binom{d}{d+j-i}$
$2e+3 \leq i \leq 3e+2$	0	$n \sum_{j=0}^{[\frac{3e+2-i}{2}]} \binom{d-1}{i-d+j} \binom{d}{j}$	$\binom{n}{i} - n \sum_{j=0}^{[\frac{3e+2-i}{2}]} \binom{d-1}{i-d+j} \binom{d}{j}$
$3e+3 \leq i \leq 4e+3$	0	0	$\binom{n}{i}$

Table 5.3   Error Pattern Distribution For Hard Decision Decoding Algorithm

$$+ \sum_{i=2e+3}^{3e+2} dn \sum_{j=0}^{[\frac{3e+2-i}{2}]} \binom{d-1}{i-d+j}\binom{d}{j} p^i q^{n-i}$$

$$+ \sum_{i=e+2}^{2e+2} i \left[\binom{n}{i} - n \sum_{j=0}^{[\frac{i-e-2}{2}]} \binom{d-1}{j}\binom{d}{d+j-i} \right] p^i q^{n-i}$$

$$+ \sum_{i=2e+3}^{3e+2} i \left[\binom{n}{i} - n \sum_{j=0}^{[\frac{3e+2-i}{2}]} \binom{d-1}{i-d+j}\binom{d}{j} \right] p^i q^{n-i}$$

$$+ \sum_{i=3e+3}^{4e+3} i \binom{n}{i} p^i q^{n-i} \}$$

Using the fact that the mean of the binomial distribution is given by np, this reduces, after some algebra, to

$$BER = p - \frac{1}{n} \sum_{i=1}^{e} i \binom{n}{i} p^i q^{n-i}$$

$$+ \sum_{i=e+2}^{2e+2} (d-i) \sum_{j=0}^{[\frac{i-e-2}{2}]} \binom{d-1}{j}\binom{d}{d-i+j} p^i q^{n-i} \qquad (5.2)$$

$$+ \sum_{i=2e+3}^{3e+2} (d-i) \sum_{j=0}^{[\frac{3e+2-i}{2}]} \binom{d-1}{i-d+j}\binom{d}{j} p^i q^{n-i}$$

The undetected bit error rate is also easily found and is given by

$$\text{UBER} = d \sum_{i=e+2}^{2e+2} \sum_{j=0}^{[\frac{i-e-2}{2}]} \binom{d-1}{j}\binom{d}{d+j-i} p^i q^{n-i}$$

$$+ d \sum_{i=2e+3}^{3e+2} \sum_{j=0}^{[\frac{i-e-2}{2}]} \binom{d-1}{i-d+j}\binom{d}{j} p^i q^{n-i}$$

(5.3)

For our last performance parameter, namely the bit deletion rate, we obtain, after some simplification,

$$\text{DBDR} = \frac{k}{n} \left\{ 1 - \sum_{i=0}^{e} \binom{n}{i} p^i q^{n-i} \right\}$$

$$-k \sum_{i=e+2}^{2e+2} \sum_{j=0}^{[\frac{i-e-2}{2}]} \binom{d-1}{j}\binom{d}{d+j-i} p^i q^{n-i}$$

(5.4)

$$-k \sum_{i=2e+3}^{3e+2} \sum_{j=0}^{[\frac{3e+2-i}{2}]} \binom{d-1}{i-d+j}\binom{d}{j} p^i q^{n-i}$$

If we assume binary coherent PSK communications over the channel, the relationship between the channel error probability p and the ratio of symbol energy E_s to noise power spectral density N_o in the channel is given by the well known formula

$$p = \frac{1}{2} - \text{erf} \sqrt{\frac{2E_s}{N_o}}$$

where

$$\text{erf}(x) = \int_0^x \frac{1}{\sqrt{2\pi}} e^{-y^2/2} dy$$

Table 5.4 lists this relationship for several values of p. In terms of the energy per data digit E_b we also have

$$\frac{E_b}{N_0} = \frac{n}{k} \frac{E_s}{N_0}$$

With the aid of these equations we may now calculate BER, UBER, and DBDR as a function of E_b/N_0. The results are shown in Tables 5.5 through 5.10, for n=7, 15, 31, 63, 127 and 225.

Figure 5-4 is a plot of logBER vs (E_b/N_0) in decibels for the maximal length shift register codes using hard decisions and the bounded distance decoding algorithm. The characteristic steep threshold behavior is evident. Thus for $E_b N_0$ above approximately 10 dB the bit error rate starts to fall off very rapidly. This threshold corresponds to the limiting performance capability of these codes in a binary symmetric channel and is consistent with the asymptotic performance bounds derived in the next section. As a point of contrast, it should be observed that the best codes operating with soft decisions over the additive, white Gaussian noise (AWGN) channel would have threshold at -1.6 dB (the Shannon limit of the AWGN channel). Indeed these very same codes when soft decision decoded with a bank of correlators do, in fact, approach the performance specified by the Shannon

Table 5.4 Channel Error Probabilities as
a Function of E_s/N_o

P	10 log E_s/N_o
.02	3.23
.04	1.87
.06	0.83
.08	-0.04
.1	-0.82
.12	-1.62
.14	-2.34
.16	-3.03
.18	-3.76
.2	-4.42
.22	-5.27
.24	-5.99
.26	-6.84
.28	-7.74
.3	-8.69
.32	-9.57
.34	-10.55
.36	-11.88
.38	-13.18
.4	-14.71
.42	-16.99
.44	-18.93
.46	-23.01
.48	-27.45
.5	$-\infty$

p	$(E_b/N_o)db$	BER	UBER	DBER
.02	6.95	2.31266E-03	1.18702E-04	3.27806E-03
.04	5.53	8.90664E-03	8.79923E-04	1.19316E-02
.06	4.53	1.92803E-02	2.74957E-03	2.44137E-02
.08	3.59	3.29504E-02	6.02955E-03	3.94469E-02
.1	2.95	.049456	1.00864E-02	5.59099E-02
.12	2.11	6.035BAE-02	1.7376AE-02	7.32066E-02
.14	1.34	8.92452E-02	2.54706E-02	9.04383E-02
.16	0.67	.111727	3.50701E-02	.107179
.18	-0.05	.135445	4.60283E-02	.12305
.2	-0.84	.160064	5.81632E-02	.137785
.22	-1.49	.185281	7.12711E-02	.151205
.24	-2.31	.21082	8.51366E-02	.16321
.26	-3.21	.236436	9.95417E-02	.173757
.28	-4.06	.261913	.114273	.182955
.3	-5.01	.287064	.129125	.190549
.32	-5.89	.311733	.143907	.196916
.34	-6.89	.335792	.158443	.202053
.36	-8.20	.359141	.172574	.206076
.38	-9.50	.381711	.186157	.209109
.4	-11.03	.403456	.199066	.211238
.42	-13.31	.424359	.211189	.21275
.44	-15.25	.444424	.222429	.213633
.46	-19.33	.463633	.232697	.214094
.48	-23.77	.482185	.241912	.214262
.5	-∞	.5	.25	.214236

Table 5.5 Bit Error Rates and Bit Deletion Rates for
Maximal Length Sequence Codes Over BSC

n=7

- 78 -

P	(E_b/N_o)dB	HER	UBER	DBDR
.02		4.98043E-05	1.18381E-06	4.82096E-05
.04		6.80538E-04	3.11892E-05	6.37658E-04
.06		2.93868E-03	1.94418E-04	2.66543E-03
.08		7.91344E-03	6.70493E-04	6.94837E-03
.1		1.64458E-02	1.66955E-03	1.39801E-02
.12		2.90065E-02	3.37962E-03	2.38742E-02
.14		4.56822E-02	5.92528E-03	3.64095E-02
.16		6.62247E-02	9.34503E-03	5.11191E-02
.18		9.01325E-02	1.35881E-02	6.73914E-02
.2		.11674	1.85268E-02	.08456
.22		.145304	2.39797E-02	.101977
.24		.175077	2.97395E-02	.119061
.26		.205362	3.56015E-02	.135327
.28		.235553	4.13874E-02	.150402
.3		.265156	4.69637E-02	.16402
.32		.2938	.0522251	.176015
.34		.321235	5.72262E-02	.186309
.36		.347317	6.19161E-02	.194893
.38		.372	6.63859E-02	.201816
.4		.395312	7.07228E-02	.207172
.42		.417341	7.50176E-02	.211085
.44		.438216	7.93469E-02	.213706
.46		.45809	8.37565E-02	.215205
.48		.477132	8.82484E-02	.215769
.5		.495514	9.27735E-02	.215593

Table 5.6 Bit Error Rates and Bit Deletion...(etc)
N = 15

Table 5.7 Bit Error Rates and Bit Deletion Rates for Maximal Length Sequence Codes Over the BSC. $n = 31$

P	(E_b/N_o)db	BER	UBER	DiDR
.02	11.19	3.47954E-08	6.10558E-11	2.15653E-08
.04	9.77	5.92089E-06	2.02454E-08	3.63631E-06
.06	8.77	1.00446E-04	5.02564E-07	6.10782E-05
.08	7.83	6.61589E-04	4.29645E-06	3.97930E-04
.1	7.19	2.59140E-03	2.04467E-05	1.54005E-03
.12	6.35	7.30123E-03	6.70525E-05	4.28179E-03
.14	5.58	1.63835E-02	1.69787E-04	9.46746E-03
.16	4.91	3.11269E-02	3.55272E-04	1.76948E-02
.18	4.19	5.21004E-02	6.41766E-04	2.90867E-02
.2	3.40	7.90193E-02	1.03108E-03	4.32219E-02
.22	2.75	.110761	1.50504E-03	5.92326E-02
.24	1.93	.145724	2.02778E-03	.076008
.26	1.03	.182148	2.55313E-03	9.24272E-02
.28	0.18	.210432	3.03512E-03	.10755
.3	-0.77	.253349	3.43896E-03	.120729
.32	-1.65	.286131	3.75017E-03	.131641
.34	-2.63	.316449	3.98013E-03	.140248
.36	-3.96	.344326	4.16703E-03	.146722
.38	-5.26	.370018	4.37184E-03	.151362
.4	-6.79	.3939	4.66966E-03	.154511
.42	-9.07	.416378	5.13756E-03	.155505
.44	-11.01	.437823	5.84042E-03	.157639
.46	-15.09	.458546	6.81718E-03	.15815
.48	-19.53	.478784	8.06947E-03	.158239
			9.55533E-01	

P	(E_b/N_o)db	BER	UBER	DBDR
.02	13.48	2.50927E-14	9.57489E-20	9.37501E-15
.04	12.06	6.67575E-10	5.02313E-15	2.48360E-10
.06	11.06	1.75643E-07	1.96535E-12	6.50250E-08
.08	10.12	6.92861E-06	1.03035E-10	2.55052E-06
.1	9.48	9.61083E-05	1.70970E-09	3.51435E-05
.12	8.64	6.85660E-04	1.54137E-08	2.40756E-04
.14	7.87	3.08388E-03	6.15978E-08	1.10841E-03
.16	7.20	9.80301E-03	3.01761E-07	3.51262E-03
.18	6.48	2.44495E-02	8.46307E-07	8.57378E-03
.2	5.69	4.93829E-02	1.90284E-06	1.70389E-02
.22	5.04	8.40962E-02	3.56669E-06	2.87257E-02
.24	4.22	.128328	5.73281E-06	4.24152E-02
.26	3.32	.175161	8.06755E-06	5.63017E-02
.28	2.47	.2209	1.00954E-05	6.86487E-02
.3	1.52	.262047	1.13650E-05	7.85188E-02
.32	0.64	.29763	1.16120E-05	8.55205E-02
.34	-0.34	.327069	1.08396E-05	9.00264E-02
.36	-1.67	.353946	9.29299E-06	9.26595E-02
.38	-2.97	.377219	7.35309E-06	9.40614E-02
.4	-4.50	.398825	5.41467E-06	9.47434E-02
.42	-6.78	.419545	3.80447E-06	9.50467E-02
.44	-8.72	.439838	2.77043E-06	9.51699E-02
.46	-12.80	.459948	2.53641E-06	9.52159E-02
.48	-17.24	.479985	3.39046E-06	9.52313E-02
.5	-∞	.499996	5.75799E-06	

Table 5.8 Bit Error Rates and Bit Deletion Rates for
Maximal Length Sequence Codes Over the BSC
N = 63.

ρ	$(E_b/N_o)db$	BER	UBER	DBDR
.02	15.86	1.87298E-26	1.50700E-37	4.08921E-27
.04	14.44	1.21539E-17	1.96921E-28	2.64753E-18
.06	13.44	7.66181E-13	1.91314E-23	1.66455E-13
.08	12.50	1.07749E-09	3.76413E-20	2.33346E-10
.1	11.86	1.85769E-07	8.69219E-18	4.00766E-09
.12	11.02	8.39112E-06	5.14012E-16	1.80174E-06
.14	10.25	1.49039E-04	1.18497E-14	3.18145E-05
.16	9.58	1.32033E-03	1.35582E-13	2.81458E-04
.18	8.86	6.96829E-03	9.01930E-13	1.46250E-03
.2	8.07	2.40603E-02	3.07377E-12	4.90904E-03
.22	7.42	5.95632E-02	1.15542E-11	1.21462E-02
.24	6.60	.113085	2.52119E-11	.022567
.26	5.70	.174755	4.10146E-11	3.39009E-02
.28	4.85	.231899	5.42363E-11	4.33945E-02
.3	3.90	.277202	5.62105E-11	4.96534E-02
.32	3.02	.311021	4.73438E-11	5.29856E-02
.34	2.04	.337026	3.29034E-11	5.44176E-02
.36	0.71	.359175	1.80797E-11	5.49248E-02
.38	-0.59	.379809	9.09001E-12	5.50734E-02
.4	-2.12	.399965	3.67961E-12	5.51098E-02
.42	-4.40	.419990	1.25634E-12	5.51172E-02
.44	-6.34	.440001	3.62523E-13	5.51182E-02
.46	-10.42	.460004	8.94962E-14	5.51186E-02
.48	-14.86	.480004	2.70661E-14	5.51186E-02
.5		.500003	6.04603E-14	5.51185E-02

Table 5.9 Bit Error Rates and Bit Deletion Rates for Maximal Length Sequence Codes Over the BSC

n = 127

- 82 -

Table 5.10 Bit Error Rates and Bit Deletion Rates for Maximal Length Sequence Codes Over the BSC

n = 255

p	$(E_b/N_0)db$	BER	UBER	BDDR
.02	18.30	1.48520E-50	2.50342E-73	1.05476E-51
.04	16.88	5.72262E-33	2.02676E-55	7.13701E-34
.06	15.88	2.06470E-23	1.21157E-45	2.57170E-24
.08	14.94	3.67366E-17	3.34488E-39	4.56801E-10
.1	14.30	9.71927E-13	1.35302E-34	1.20602E-13
.12	13.46	1.74272E-09	3.71641E-31	2.15679E-10
.14	12.69	4.75921E-07	1.55955E-28	5.87005E-08
.16	12.02	3.21466E-05	1.61169E-26	3.94710E-06
.18	11.30	7.36676E-04	5.55355E-25	8.98905E-05
.2	10.51	7.14795E-03	7.81977E-24	8.64701E-04
.22	9.86	3.47472E-02	5.19014E-23	4.15003E-03
.24	9.04	9.74022E-02	1.90091E-22	1.14125E-02
.26	8.14	.179573	3.52998E-22	2.04466E-02
.28	7.29	.248313	4.14043E-22	2.71464E-02
.3	6.34	.291426	3.04193E-22	3.02423E-02
.32	5.46	.318412	1.44862E-22	3.11647E-02
.34	4.48	.339001	4.59159E-23	3.13465E-02
.36	3.15	.359987	9.88358E-24	3.13707E-02
.38	1.85	.380003	1.46646E-24	3.13729E-02
.4	0.32	.400007	1.51570E-25	3.13731E-02
.42	-1.96	.420008	1.09877E-26	3.13732E-02
.44	-3.90	.440004	5.60717E-28	3.13729E-02
.46	-7.98	.460008	2.01618E-29	3.13731E-02
.48	-12.42	.480006	5.12318E-31	3.13731E-02
.5	- ∞	.500006	1.42572E-31	3.13729E-02

Figure 5.4 Bit Error Rate for N=7,31,63,127,255,512
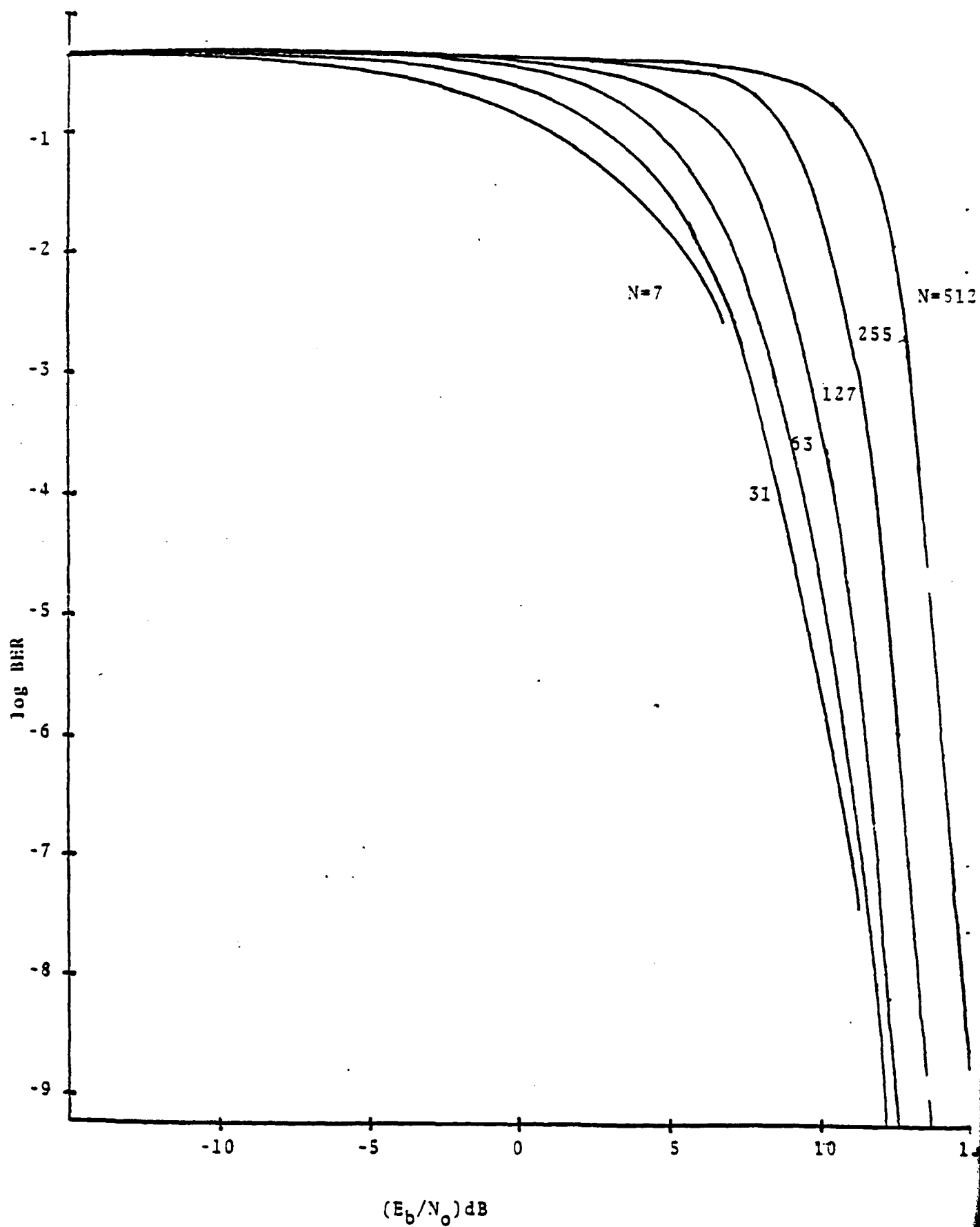
limit. This is discussed in section 2.4 and the corresponding curves are shown in Figure 5-9.

On the basis of these tables and curves we make the following observations:

1) If we arbitrarily define an acceptable bit error rate as $BER \leq 10^{-5}$ and an acceptable data bit deletion rate as 1% of the data transmitted, i.e., $DBDR \leq 0.01$ k/n, then the minimum required value of E_b/N_o increases with code length from 7db at n=7 to 11 db at n=225.

2) For the most practical cases of n=127 and n=255 the undetected bit error rate UBER is essentially zero, for all values of E_b/N_o. Thus, even though the error correcting ability of the decoder becomes marginal at E_b/N_o less than 7db, virtually none of the residual decoder output errors go undetected. This feature appears to be the one of major importance in the application of the maximal length sequence codes to spread spectrum systems.

2.4 Asymptotic Performance Bounds

For large values of n the Plotkin bound on minimum distance for a binary linear block code takes the asymptotic form (1)

$$\frac{k}{n} \leq 1 - 4 \frac{d}{2n}$$

Letting $e = 2d - 1$ can be the code's error correcting capability, this can be rewritten as

$$. \; e \leq \frac{n-k}{4}$$

The implication of this formula is that the error correcting capability of a bounded distance hard decision decoding algorithm for sufficiently long linear (n,k) block codes is limited to one fourth the number of parity digits in a codeword.

If this code is used on a binary symmetric channel with bit error rate

$$p > \frac{1}{4}(\frac{n-k}{n}) \quad ,$$

the code is unable to correct even the average number of errors np in a codeword. Thus, in an absolute sense of the upper limit on the correctable channel bit error rate is given by

$$P_{max} \leq \frac{n-k}{4n}$$

For the maximal length sequence codes n-k~n as n becomes large and $P_{max} \sim 1/4$. This is equivalent to $E_s/N_o = 6.25$ db, which represents the minimum channel signal-noise ratio for which the

maximal length sequence codes are useful under a hard decision decoding rule, using BER as the criterion of performance.

Let us now derive an asymptotic expression for BER in the limit as n becomes large, assuming p<1/4. Without difficulty we may show that the double sum in equation (5.2) is then negligible, so that

$$BER \approx p - \frac{1}{n} \sum_{i=1}^{e} i \binom{n}{i} p^i q^{n-i}$$

Since

$$\sum_{i=1}^{n} i \binom{n}{i} p^i q^{n-i} = np$$

this can be rewritten in the form

$$BER \approx \frac{1}{n} \sum_{i=e+1}^{n} i \binom{n}{i} p^i q^{n-i}$$

If we then let $i-1 \triangleq r$, then

$$BER = p \sum_{r=e}^{n-1} \binom{n-1}{r} p^r q^{n-r-1}$$

The sum is now the "tail" of a binomial distribution which we can overbound. From Peterson and Weldon [1] we have

$$\sum_{\lambda(n-1)}^{n-1} \binom{n-1}{r} p^r q^{n-r-1} \leq (\frac{p}{\lambda})^{\lambda(n-1)} (\frac{q}{\mu})^{\mu(n-1)}$$

where $u \triangleq 1-\lambda$

provided $\lambda > p$. [This is the same as saying that the sum starts to the right of the mean of distribution.]

For the codes in question,

$$e = \frac{n-3}{4} \triangleq \lambda(n-1)$$

or

$$\lambda = \frac{\frac{n-3}{4}}{n-1} \rightarrow \frac{1}{4}$$

as $n \rightarrow \infty$. Hence for all $p < 0.25$ the above inequality holds asymptotically. Then substituting we have

$$BER \leq p \left(\frac{p}{\lambda}\right)^{\lambda(n-1)} \left(\frac{q}{\mu}\right)^{\mu(n-1)}$$

for large n, $\lambda \rightarrow \frac{1}{4}$ and $\mu \rightarrow \frac{3}{4}$

and

$$BER \leq p \left(9.48 \; pq^3\right)^{\frac{n-1}{4}} \qquad (5.5)$$

as an asymptotic upper bound on BER for all $p < 1/4$.

Next we establish approximations to BER, UBER, and DBDR valid for values of $p \leq 0.02$. In this case the first terms of equations (5.2),(5.3) and (5.4) suffice and we have

$$BER \approx \binom{n-1}{e} p^{e+1} q^{n-e-1}$$

$$\text{UBER} \approx d\binom{d}{e} p^{e+2} q^{n-e-2} \tag{5.6}$$

$$\text{DBDR} \approx \frac{k}{e+1}\binom{n-1}{e} p^{e+1} q^{n-e-1}$$

As an example of the quality of these approximations we evaluate equations (5.6) for the p=0.02 and n=7. Then

$$\text{BER} \approx 6p^2 q^5 = 2.17 \times 10^{-3}$$

$$\text{UBER} \approx 16p^3 q^4 = 1.18 \times 10^{-4}$$

$$\text{DBDR} \approx 9p^2 q^5 = 3.25 \times 10^{-3}$$

Note the excellént agreement with the exact values of Table 5.5.

2.5 Correlation Decoding

With the fundamental limitations of maximal length sequence codes with hard decision decoding well established, we turn our attention to their performance under a correlation decoding regime.

We assume that each of the n-1 maximal length sequence codewords $u_1, u_2, \ldots u_{n-1}$ is composed of n elemental antipodal signals $s_o(t)$, $s_1(t)$, where $0 \le t < T/n$ and T is the duration of the codeword. The set of codewords then forms a so called transorthogonal or regular simplex code with correlation coefficient -1/(n-2) between unlike codewords (2).

Let the codewords be transmitted with equal probability and received in white Gaussian noise of power spectral density
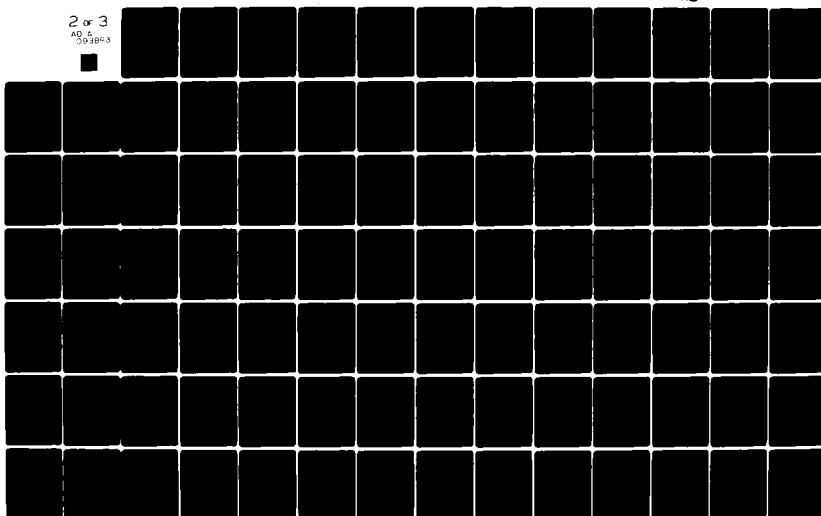
N_o watts per hertz. With coherent reception the optimum decoder is then easily shown to take the form indicated in Figure 5.9. Here v(t) is the received signal corresponding to the transmission of a codeword and the maximization of the outputs from the n-1 integrators takes place at the end of the time period T.

We may also show that under this decoding scheme the code bit error rate is minimum and given by

$$BER = \frac{n-1}{2n} \left\{ 1 - \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-x^2/2} \left[erfc(x + \frac{kE_b(n-1)}{N_o(n-2)}) \right]^{n-2} dx \right\}$$

where E_b is the energy per information bit and erfc(x) is defined by

$$erfc(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-y^2/2} dy$$

Figure 5.9 shows BER as obtained from a numerical evaluation of the above formula (2), as a function of E_b/N_o, for a number of values of k. If we again take 10^{-3} as the maximum acceptable value of BER, then the minimum required E_b/N_o decreases with increasing k from 4 db at k=3 to 2 db at k=8. Comparing these values with Section 2.4 it is clear that the correlation decoding significantly improves upon the performance of these codes over the Gaussian channel. The complexity of implementation, however, is likely to be much greater than for hard decision schemes.
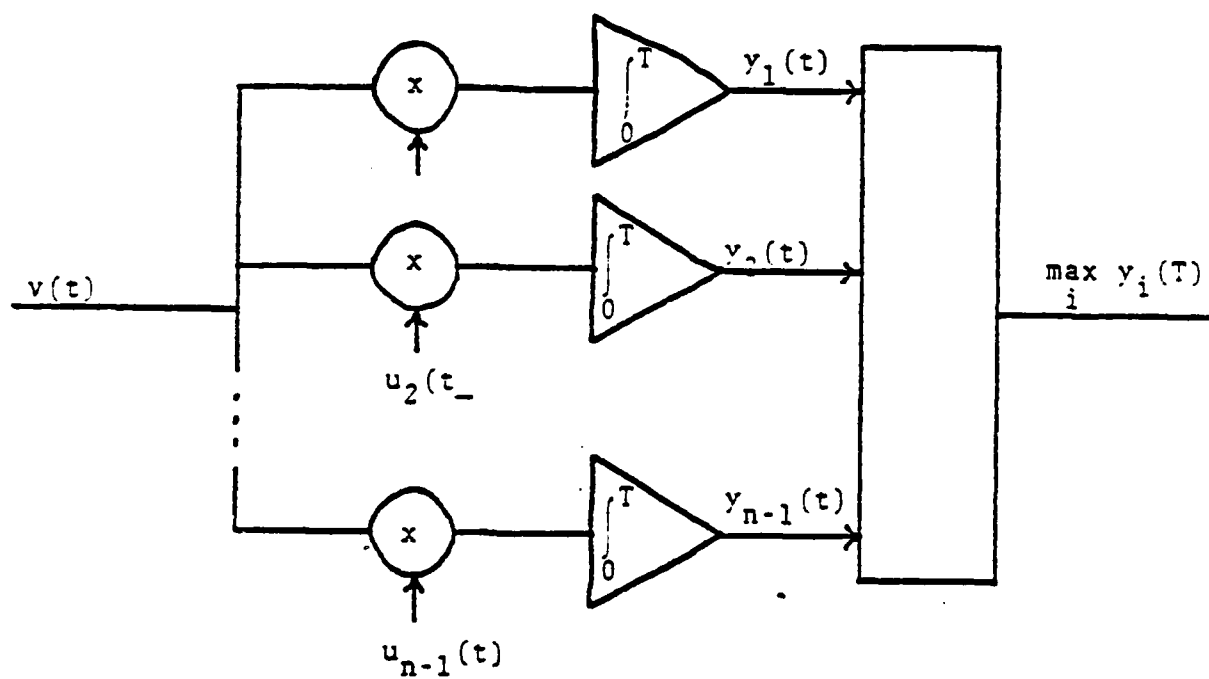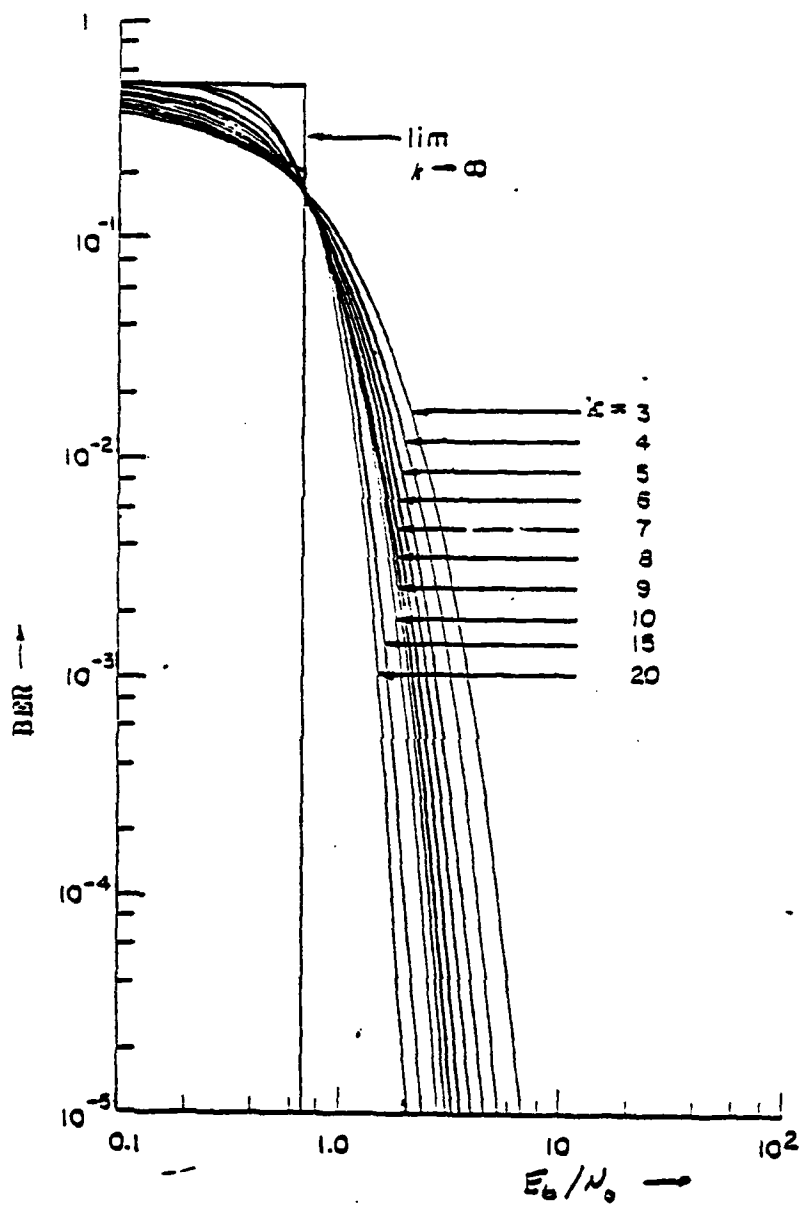
Figure 5.8 Correlation Decoder

Figure 5.9 BER for Correlation Decoding of Maximal Length
Sequence Codes

Section 3. Low Rate Convolutional Codes

3.1 Code Definition and Properties

A binary convolutional code of rate R = 1/n and constraint
length k is a mapping of a semi-infinite sequence of data
digits into a semi-infinite sequence of channel digits such
that each data digit corresponds to n channel digits which are
functions of the last k data digits. The implementation of
such a coding scheme is accomplished in terms of a socalled for-
ward shift register consisting of k stages, from which n sub-
sets of the k stages are connected to modulo - 2 adders. The
outputs from these adders are then sampled once per data digit
and transmitted over the channel.

Letting 1 and 0 denote connection or no connection, respec-
tively, of a given stage to a given modulo - 2 adder, the code
is completely defined by a set of n binary k-tuples, the so-
called code generators $G_1, G_2 ... G_n$.

As an example, consider the case k=3. One possible form
of the encoder is shown in Figure 5.10. Clearly, the two
generators are G_1 =

$$G_1 = 111 \quad ; \quad G_2 = 101$$

and the code has rate R-1/2.

In general, if we insist on each generator being distinct,
then n is limited to $n \leq 2^k - 1$. In the remainder of this section
we consider the case $n = 2^k - 1$, so that the generators are all
binary nonzero k-tuples. The rate of these codes is then
$R = 1/(2^k - 1)$ and the channel bandwidth is expanded by a factor
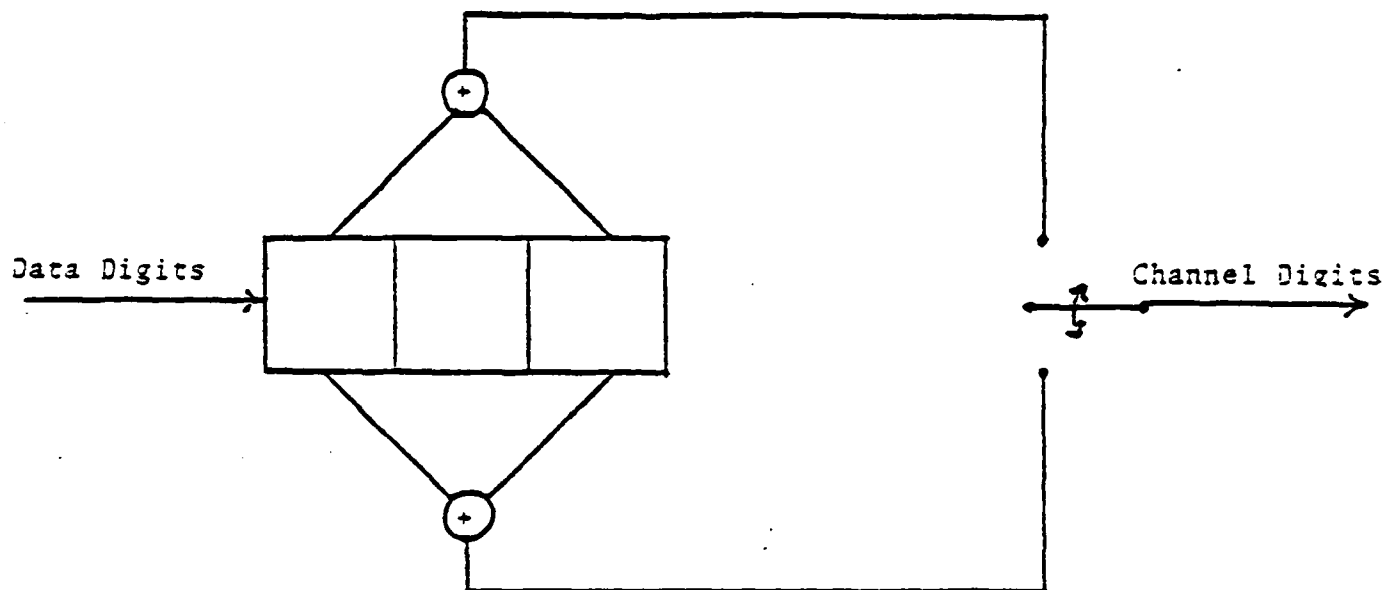
Figure 5.10 Binary Convolutional Code of
           Rate of 1/2 and Constraint Length 3

of 2^k-1.

In Table 5.11 we have listed the set of generators as the rows of an array. Without great difficulty we may show that except for the absence of all zero k-tuple, the rows of the array form a vector space of dimension n over the binary field. Each column and each modulo-2 sum of two or more columns contains exactly 2^{k-1} ones and $2^{k-1}-1$ zeros. In fact, the columns can be thought of as the basis vectors for a vector space of dimension k which is exactly the set of codewords of the (n,k) maximal length sequence code discussed in Section 2.

$$\underline{c_1 \; c_2 \; c_3 \; . \; . \; . \; . \; . \; c_k}$$

$$1 \quad 0 \quad 0 \; . \; . \; . \; . \; 0 \; 0$$

$$0 \quad 1 \quad 0 \; . \; . \; . \; . \; 0 \; 0$$

$$0 \quad 0 \quad 1 \; . \; . \; . \; . \; 0 \; 0$$

$$.$$

$$.$$

$$.$$

$$.$$

$$0 \quad 0 \quad 0 \; . \; . \; . \; . \; 0 \; 0$$

$$.$$

$$.$$

$$.$$

$$.$$

$$1 \quad 1 \quad 1 \; . \; . \; . \; . \; 1 \; 1$$

Table 5.11   Generators of $n = 2^k-1$ Convolutional Code

Under the usual definition for convolutional codes [1], the general form of the $k \times k(2^k-1)$ generator matrix G is given by

$$G = \begin{bmatrix} c_1^T & c_2^T & c_3^T & \cdots & & & & c_k^T \\ & c_1^T & c_2^T & & \cdots & & & c_k^T \\ & & \ddots & & & & & \\ & & & \ddots & & & & \\ & & & & \ddots & & & \\ & & & & & \ddots & & \\ & & & & & & \ddots & c_1^T \end{bmatrix}$$

where c_i^T is the transpose of the ith column of Table 5.11 and the empty places in G contain all zeros.

Since the first digit of c_1 is always one and the first digits of $c_2, c_3, \ldots c_k$ are always zero, the code is clearly systematic. The parity check matrix is therefore given by

$$H = \begin{bmatrix} \bar{c}_1 & I \\ \bar{c}_2 & 0 & \bar{c}_1 & I & \cdot \\ \bar{c}_3 & 0 & \bar{c}_2 & 0 & & \cdot \\ \cdot & & \cdot & & & & \cdot \\ \cdot & & \cdot & & & & & \cdot \\ \cdot & & \cdot & & & & & & \cdot \\ \bar{c}_k & 0 & \bar{c}_{k-1} & 0 & \cdots & & & & & \bar{c}_1 & I \end{bmatrix}$$

where $\bar{c_i}$ is equal to c_i with the first digit deleted and I and 0 are unit matrices and zero matrices of size 2^k-2 x 2^k-2.

As an example, consider the case k=3. Then the generators in array form are

c_1	c_2	c_3
1	0	0
0	1	0
0	0	1
1	1	0
1	0	1
0	1	1
1	1	1

$$G = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

and its parity check matrix takes the form

$$
H = \left[
\begin{array}{c:c:c:c}
\begin{array}{c} 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \end{array} & I_{6\times6} & 0_{6\times7} & 0_{6\times7} \\
\hdashline
\begin{array}{c} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{array} & 0_{6\times6}\ \begin{array}{c} 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ \end{array} & I_{6\times6} & 0_{6\times7} \\
\hdashline
\begin{array}{c} 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \end{array} & 0_{6\times6}\ \begin{array}{c} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{array} & 0_{6\times6}\ \begin{array}{c} 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \end{array} & I_{6\times6}
\end{array}
\right]
$$

where $[x]$ denotes the interger part x.

From the standpoint of performance, the most important parameter of a convolutional code is its free distance d_{free}, defined as the number of ones in that semi-infinite channel sequence with the smallest nonzero number of ones. We now derive this quantity for the codes defined above.

Suppose the data sequence to be encoded consists of a leading one, followed by an arbitrary semi-infinite sequence of ones and zeros. Then as the leading one enters the leftmost stage of the encoding register and traverses to the last stage, the successive channel n-tuples correspond to sums of columns of the array in Table 5.11 and are hence codewords of a maximal length sequence code. The total number of ones in each output n-tuple is therefore always 2^{k-1} and since the leading one traverses k stages, the output sequence corresponding to a semi-infinite input sequence must have at least $k2^{k-1}$ ones. Consequently,

$$d_{free} \geq k \; 2^{k-1}$$

On the other hand, for the data sequence

$$1 \; 0 \; 0 \; . \; . \; . \; . \; . \; 0 \; . \; . \; . \; . \; .$$

the output sequence has exactly $k \; 2^{k-1}$ ones and we have

$$d_{free} = k \; 2^{k-1} \tag{5.7}$$

Next, we compare d_{free} in (5.7) with a well known upper bound (3) which for codes of rate $R=1/(2^k-1)$ we may state as follows

$$d_{free} \leq \min_{j \geq 1} \left[(\frac{2^k-1}{2})(\frac{2^j}{2^j-1})(k+j-1) \right] \tag{5.8}$$

We note that for large values of k this bound approaches $k2^{k-1}$, so that the codes defined above have maximum free distance for large constraint lengths.

In Table 5.12 we compare equations (5.7) and (5.8) for $k\leq 10$. Also included are the free distances of good $R=1/(2^k-1)$ codes constructed from optimum $R=1/2$, $R=1/3$ and $R=1/4$ codes by generator replication [3].

Another important parameter of performance is the code's minimum distance, defined in the usual way as the minimum distance among k successive blocks of encoder output digits corresponding to k successive encoder input digits corresponding to k successive encoder input digits of which the first is a one. Since each block of 2^k-1 decoder outputs is a codeword in a maximal length sequence code, the distance between such blocks is exactly 2^{k-1}. Thus we have immediately

$$d_{min} = k\,2^{k-1} = d_{free}$$

and the code can therefore correct all patterns of up to

$$e = k\,2^{k-2}-1$$

errors among any successive $k(2^k-1)$ channel digits.

Although the calculation of a convolutional code's performance over a given channel and under a specific decoding algorithm is generally a difficult problem, good upper bounds on the basic bit error rates can readily be obtained if the so-called generator function of the encoder state transitions is known. We now turn our attention to a derivation of this quantity.

K	max d_{free}	$h2^{k-1}$	constructable d_{free}
3	18	12	18
4	50	32	50
5	124	80	124
6	288	192	285
7	653	448	635
8	1457	1024	1402
9	3212	2304	3066
10	7014	5120	6905

Table 5.12 Free Distance Comparisons of
$R=1/(2^k-1)$ Convolutional Codes

Since the calculations involved are somewhat tedious, we restrict ourselves to the case k=3 for purposes of illustration and merely state the result for general values of k.

Let the states of the encoding register of Figure 5.6 correspond to the four combinations 00,01,10,11 of the contents of the last two register stages and denote these states by x_1, x_2, x_3, x_4, respectively. Then it is clear that under a zero input the states undergo the transitions

$$x_1 \to x_1'$$
$$x_2 \to x_1$$
$$x_3 \to x_2$$
$$x_4 \to x_2$$

where x_1' is the zero state reached from a nonzero state. Likewise, under a one input we have

$$x_1 \to x_3$$
$$x_2 \to x_3$$
$$x_3 \to x_4$$
$$x_4 \to x_4$$

Using standard signal flow graph techniques we now reason as follows:

For every state transition the output from the encoder is always a codeword from the maximal length sequence code and so has weight 4, except for the transition $x_1 \to x_1$. The input to the encoder corresponding to a given state transition is, of course, either zero or one. Let the exponents of the variables D and N correspond to the weights of the output and input, respectively.

- 102 -

(E_b/N_o)dB

- 103 -
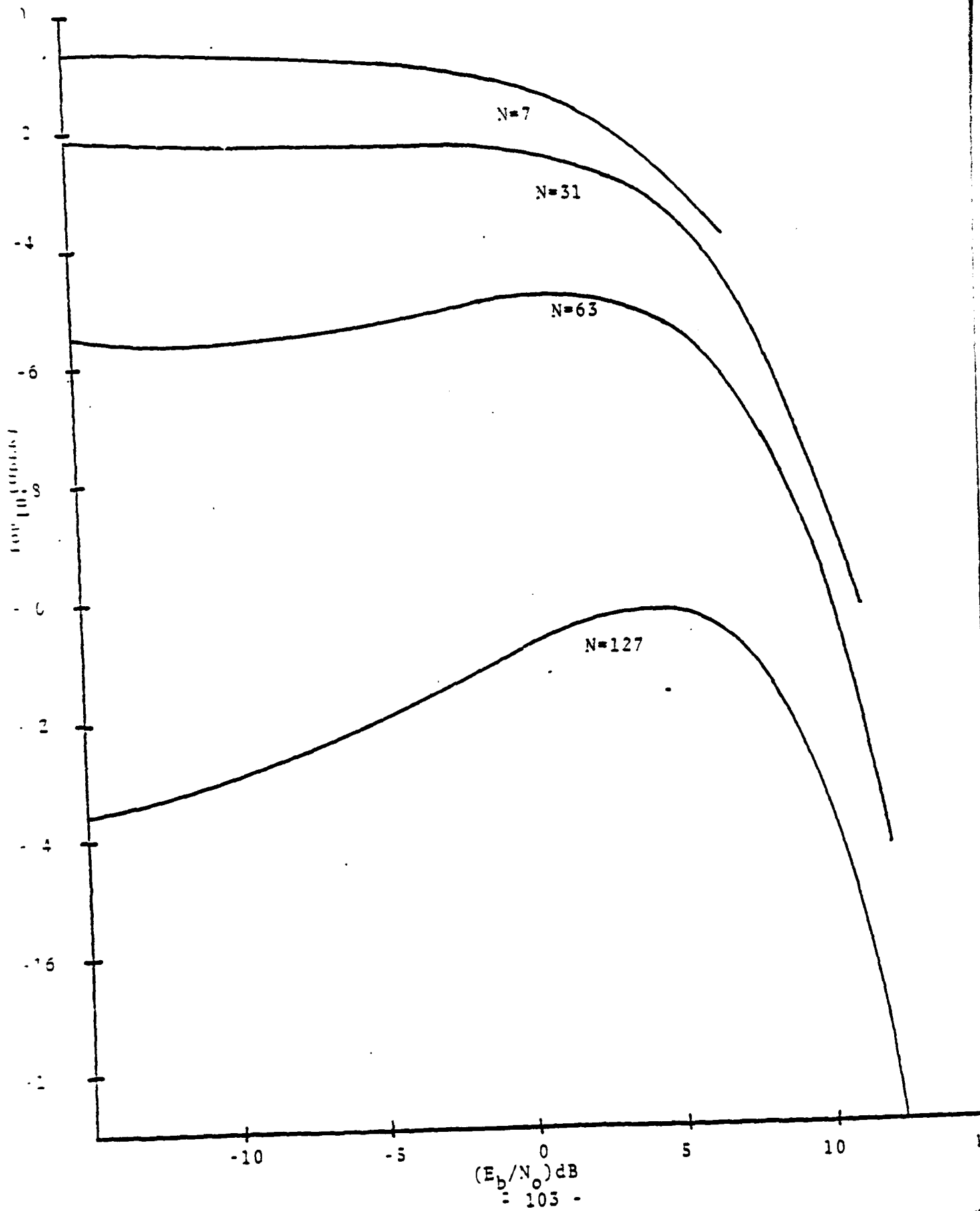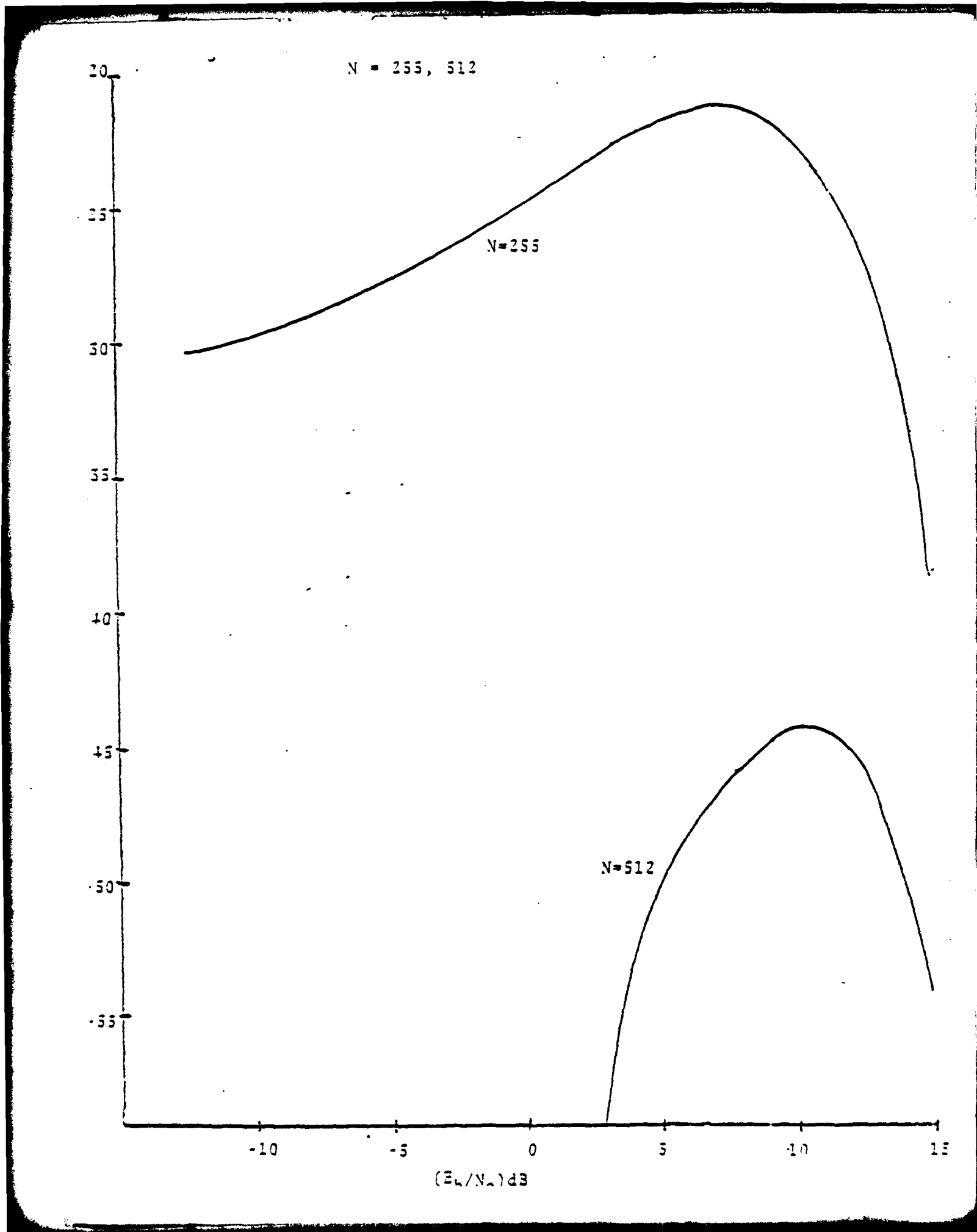
N = 255, 512

Figure 5.7 Data Bit Deletion Rate (DBDR) For
N=7,31,65,127,255,512
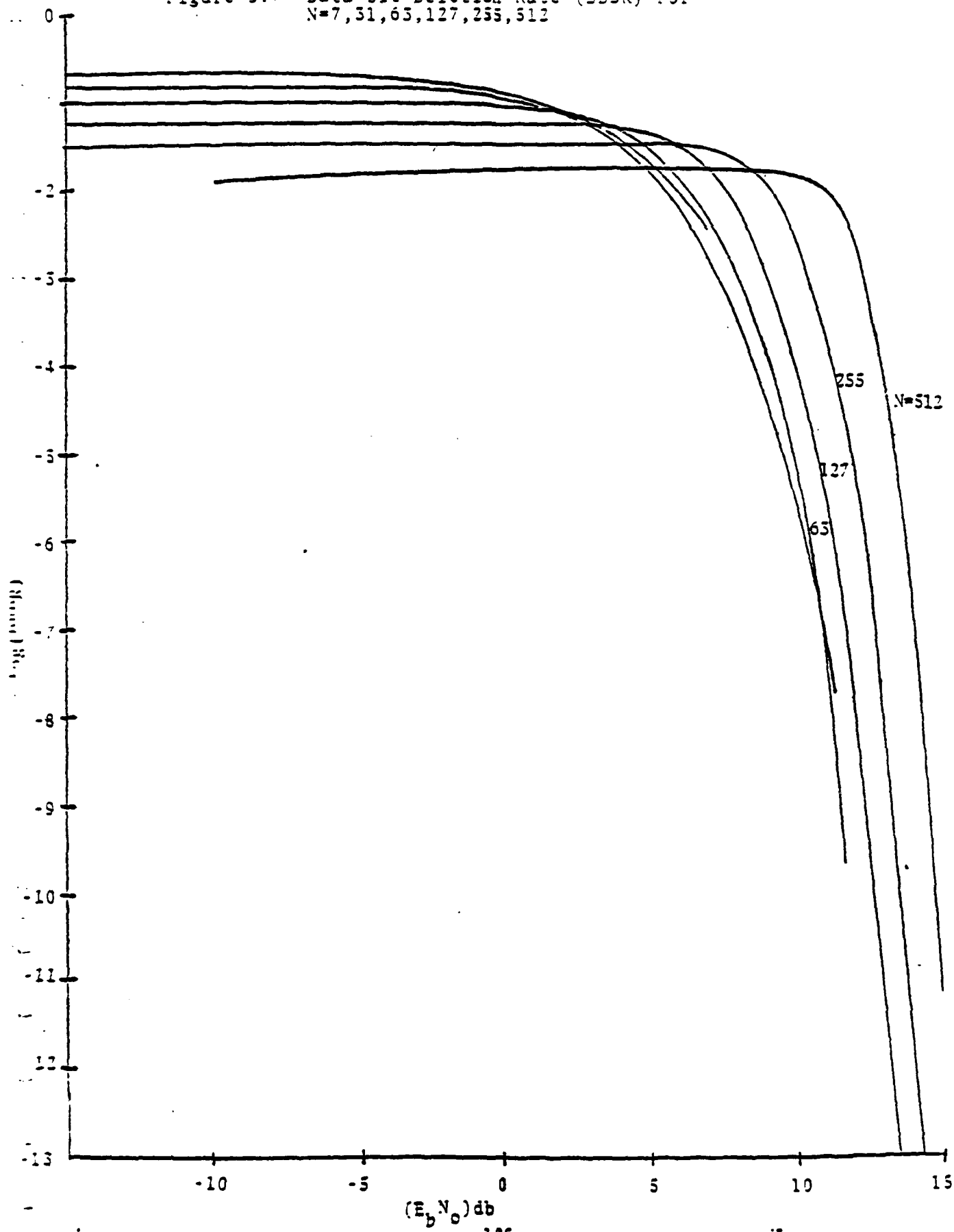
Then summing at the four states we have

$$x_1' = D^4 x_2$$
$$x_2 = D^4 (x_3 + x_4)$$
$$x_3 = D^4 N (x_1 + x_2)$$
$$x_4 = D^4 N (x_3 + x_4)$$

Solving these four equations in five variables for the ratio x_1'/x_1 yields the generator function

$$T(D,N) = \frac{ND^{12}}{1 - ND^4 - ND^8}$$

In the general case a similar technique leads to the form

$$T(D,N) = \frac{ND^{k2^{k-1}}}{1 - N \sum_{i=1}^{k-1} D^{i2^{k-1}}} \tag{5.9}$$

The derivative of the generator function with respect to N and evaluated at $N=1$ is also of interest and is given by

$$\left. \frac{dT(D,N)}{dN} \right|_{N=1} = \frac{D^{k2^{k-1}}}{(1 - \sum_{i=1}^{k-1} D^{i2^{k-1}})^2} \tag{5.10}$$

5.2 Performance Bounds

With the aid of equations (5.7) and (5.8) we may easily
obtain upper bounds on the residual bit error rate when the above
codes are used for signalling over the additive white Gaussian
noise channel and decoded via a maximum liklihood algorithm.
The pertinent formulas are (4):

    a. For hard decision decoding:

$$P_b < \frac{dT(D,N)}{dn} \Bigg|_{N=1, \ D = \sqrt{4pq}}$$

$$= \frac{(4pq)^{k2^{k-2}}}{\left\{ 1 - \sum_{i=1}^{k-1} (4pq)^{k2^{k-2}} \right\}^2}$$

where p is the channel bit error rate and q = 1-p.

    b. For soft analog decision decoding:

$$P_b < erfc' \sqrt{2d_{free} R \frac{E_b}{N_0}} \ e^{d_{free} R \frac{E_b}{N_0}}$$

$$\times \frac{dT(D,N)}{dN} \Bigg|_{N=1, \ D=e^{-R\frac{E_b}{N_0}}}$$

$$= \frac{\text{erfc}\sqrt{2d_{free}R\frac{E_b}{N_o}}\, e^{d_{free}R\frac{E_b}{N_o}}\, e^{-Rk2^{k-1}\frac{E_b}{N_o}}}{\left\{1 - \sum_{i=1}^{k-1} e^{-Ri2^{k-1}\frac{E_b}{N_o}}\right\}^2}$$

where $R = \dfrac{1}{2^k-1}$ as the rate of the code and $\text{erfc}'(x)$ is defined by

$$\text{erfc}'(x) = \int_x^{\infty} \frac{1}{\sqrt{2\pi}}\, e^{-y^2/2}dy$$

Using (5.5), this reduces to

$$P_b < \frac{\text{erfc}'\sqrt{2d_{free}R\frac{E_b}{N_o}}}{\left\{1 - \sum_{i=1}^{k-1} e^{-Ri2^{k-1}\frac{E_b}{N_o}}\right\}^2} \qquad (5.11)$$

For finite constraint lengths, and large values of $\frac{E_b}{N_o}$ the bound of (5.11) approaches the form

$$P_b < \text{erfc}' \sqrt{2d_{free}R\frac{E_b}{N_o}}$$

We may compare this result with the case of no coding by noting that for the latter

$$P_b = \text{erfc}' \sqrt{2\frac{E_b}{N_o}}$$

Clearly then, for large $\frac{E_b}{N_o}$ the so called coding gain of the codes discussed here satifies the bound

$$\text{Coding gain} \geq d_{free}R = \frac{k2^{k-1}}{2^k-1} \to \frac{k}{2} \qquad (5.1\text{?})$$

which is a monotonically increasing function of the constraint length.

For finite $\frac{E_b}{N_o}$ the denominator of (5.11), which for reasonably large k assumes the form

$$\{1 - \sum_{i=1}^{k-1} e^{-\frac{i}{2}\frac{E_b}{N_o}}\}^2$$

must of course, be taken into account. For k=7 and E_b/N_o=3db, for example, its value is approximately 0.2. The numerator in this case equals $\text{erfc}' \sqrt{14.1} \approx 10^{-4}$, so that

$$P_b < 5\times10^{-4}$$

Note that this bound compares favorably with the performance
of the standard rate 1/2, constraint length 7 convolutional
code decoded via the soft decision Viterbi algorithm [4].

As our last special case, let $E_b/N_o \to 0$. Then the denomi-
nator of (5.11) approaches the value $(2-k)^2$. For $k \geq 3$ this is
no smaller than one and thus the codes exhibit a coding gain
which is bounded by (5.12)

5.3 Concatenated Convolutional Codes

An alternate means for obtaining a low rate convolutional
code is by means of cascading or concatenating a single moderate
rate code. For example, if the output of a rate 1/2 encoder is
fed into a rate 1/2 encoder the result is a rate 1/4 code. Then
concatenating k times can yield a rate $1/2^k$ code. If the same
code is used in each stage of the concatenation, then a single
Viterbi Algorithm decoder can be used for decoding. The first
inner decoder will give a coding gain in that the BER at its
output will be improved. If the output of the first inner
decoder is then applied to the input of the second inner decoder,
the raw bit error rate should again be improved provided, of
course, that the second decoder bit input errors behave as if
they were derived from a binary symmetric, memoryless channel.
Unfortunately, the first decoder output errors are known to be
bursty. However, it is nevertheless possible to achieve the
coding gain per stage by interleaving and deinterleaving. The
coding gain advantage of these codes have been explored in
this study but it is felt that because of the excessive delays

in the interleaving-deinterleaving process and because the
"complete" convolutional codes described in the previous section
appeared so promising, that we should postpone further inquiry
into concatenated convolutional codes.

3.4 Conclusions

We have investigated the basic properties of a class of low
rate convolutional codes for the purpose of assessing their
usefulness in direct sequence spread spectrum applications.
In summary, we note the following points:

a. Compared to the generation of PN sequences, the
2^k-1 tap configurations of the convolutional encoders imply
a somewhat larger complexity, although from the standpoint of
LSI technology this seems of little real consequence.

b. Since maximum liklihood decoding of convolutional codes
is in a sense self-synchronizing--although at the loss of some
data--the need for PN sequence synchronization at the receiver
is obviated. This appears to be a major advantage of the low
rate coding approach. Of course, bit synchronization is
required in both cases, representing the fundamental limit to
spread spectrum operation at low signal-to-noise ratios.

c. Due to their extremely long branch sequences the codes
discussed above appear to be of particular value on bursty
channels, elimating or at least reducing the need for interleaving.
The savings in storage requirements, decoding delay and inter-
leaver array synchronization--to name just a few--are likely
to be appreciable.

- 111 -

d. Unlike standard direct sequence methods, low rate coding
exhibits a coding gain beyond the normal processing gain achieved
through correlation decoding of the PN sequences. For high and
low signal-to-noise ratios this coding gain increases linearly
with constraint length and is therefore limited only by the
constraint length.

e. Compared to the alternative configuration of direct
sequence spectrum spreading in conjunction with rate 1/2, con-
straint length 7 convolutional codes decoded via the Viterbi
algorithm, it is clear that low rate coding has the advantage
of requiring only one level of implementation. Another advantage
resides in the fact that the coding gain for the same total
bandwidth expansion can be much larger than the 5 db achieved with
rate 1/2 codes. Finally, we note the possibility of better per-
formance in a bursty environment where Viterbi decoding of rate
1/2 codes requires substantial interleaving and usually results
in a reduced coding gain.

References for Chapter 5

[1] W.W. Peterson and E.J. Weldon, "Error Correcting Codes",
     MIT, 1972.

[2] A.J. Viterbi, "Principles of Coherent Communication",
     McGraw-Hill, 1966.

[3] D. Wiggert, "Error-Control Coding and Applications", Artech
     House, 1978.

[4] J.J. Spilker, "Digital Communications by Satellite",
     Prentice-Hall, 1977.

CHAPTER 6

Adaptive Null-Steering Arrays Transient Behavior

Introduction

This Chapter of the final report presents a novel approach for analyzing the transient behavior of adaptive array for both deterministic and random jammers. Using this technique the behavior of adaptive arrays in the presence of step and periodic jammers is analyzed. The analysis has been carried out for adaptive nulling antennas which have stationary or fixed main beams. This corresponds to the tactical scenario of a line of sight link in a jamming environment.

The first section of the report presents a review of the Widrow[1] and Applebaum[2,3] algorithms. These two algorithms represent the starting point of most modern adaptive theory. The Widrow or least mean square error (LMS) algorithm was developed by Widrow and his co-workers. They applied methods used in adaptive filters to phased arrays. About the same time Applebaum developed a similar algorithm based on maximizing the signal to noise ratio. The algorithm is sometimes called the MSN algorithm. Following Widrow's initial work the LMS algorithm was developed further by Griffiths[4] and Frost.[5] They found that one can maintain a chosen frequency characteristic for the array in a desired direction while discriminating against noises coming from other directions. Brennan and Reed[6,7] extended the domain of the MSN algorithm to include adaptivity both in the spatial as well as the temporal (Doppler

PRECEDING PAGE BLANK-NOT FILMED

filter) domain. Recent emphasis in the field has been on speeding up loop convergence. A number of techniques such as orthogonalization,[3] null control law modification[9] have appeared in the recent literature in this connection.

The second section of the report is devoted to a deterministic analysis of an analog implementation of the LMS algorithm. Here it is assumed that deterministic signals are incident upon the array and that receiver noise can be neglected. Three particular examples are explored: first, the case of a step plane wave normally incident upon the array; second, a step jamming signal incident obliquely on the array; and third, a periodic pulsed jamming signal obliquely incident on the array. For all three cases, an explicit solution is obtainable.

The third section of the report treats the wide spectrum random jammer applied to an array utilizing the Widrow algorithm. The approximate equation for the mean is found exactly by using special group properties of the governing equations. This exact solution leads to explicit expressions for the decay coefficients. These decay coefficients are related explicitly to the number of elements in the array, the signal amplitude and the jammer noise power. The case of a step jammer is treated in detail and certain conjectures are made about the behavior of the array when a periodic jammer is applied.

Adaptive Array Principles

A review of the principles of operation for the Widrow and Applebaum algorithms will now be presented. Although there

- 116 -

exist many variants of these two basic structures, an under-
standing of the motivation and operation of these algorithms
is fundamental.

Let us consider the Widrow algorithm first. Let $X_1(t)$,
$X_2(t), \ldots X_N(t)$ denote the quadrature outputs of an N/2 ele-
ment phased array. They are assumed to be random functions of
time. Suppose these functions are multiplied by weighting
factors $W_1, W_2, W_3, \ldots W_N$ and added to form the output

$$S(t) = \sum_{\ell=1}^{N} W_\ell X_\ell(t) \qquad (6.1)$$

as indicated in Figure 6.1.

Next $S(t)$ is compared to a reference function $R(t)$ which
may be taken as the "desired" response) with the object of mini-
mizing the error $\varepsilon(t)$, or more precisely, its mean squared, viz,

$$<|\varepsilon(t)|^2> \qquad (6.2)$$

Thus one would like to adjust the weights $W_1, W_2, W_3, \ldots W_N$ such
that

$$<|\varepsilon(t)|^2> = <|R(t) - \sum_{\ell=1}^{N} W_\ell X_\ell(t)|^2> \qquad (6.3)$$

is minimized. Both $X_\ell(t)$ and the reference function $R(t)$ (a
stochastic process) are assumed real. The case of complex
functions will be considered later. Thus (the symbol ξ will
stand for mean squared error).

$$\xi \equiv <|\varepsilon(t)|^2> \equiv <\varepsilon(t)^2>$$

- 117 -

$$S(t)=\sum_{i=1}^{N} w_i x_i(t)$$
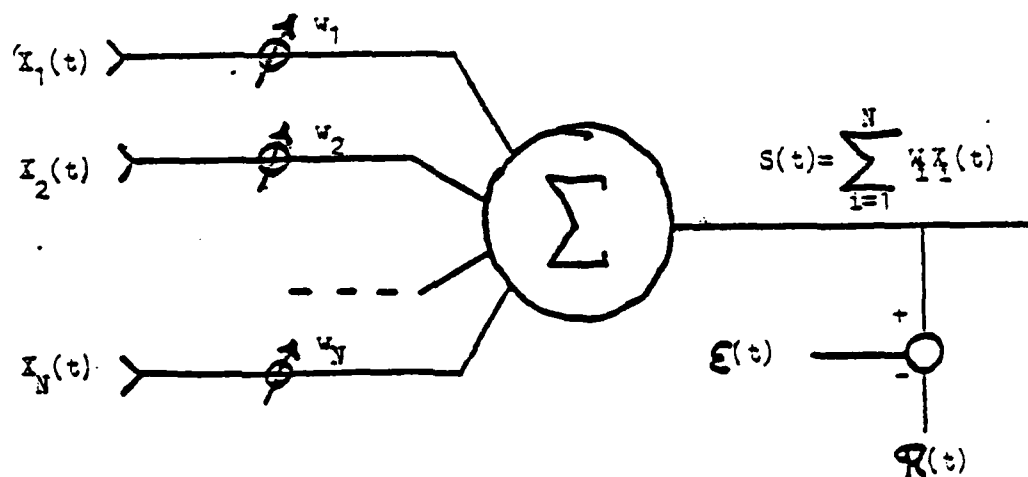
Figure 6-1   Adaptive Array

$$= <R^2(t)> - 2<R(t) \sum_{\ell=1}^{N} W_\ell X_\ell(t)> + \sum_{\ell,m} W_\ell W_m <X_\ell(t)X_m(t)>$$

Note that the weights W_ℓ are not functions of time and are taken as nonrandom parameters. Differentiating ϵ with respect to W_n and setting each derivative to zero, yields

$$0 = \frac{\partial \epsilon}{\partial W_n} = -2 <R(t)X_n(t)> + 2 \sum_{\ell=1}^{N} W_\ell <X_\ell(t)X_n(t)> \qquad (6.4)$$

$$n=1,2,\ldots,N$$

Now define column matrices

$$\underline{S} = \begin{bmatrix} <R(t)X_1(t)> \\ \cdot \\ \cdot \\ \cdot \\ <R(t)X_N(t)> \end{bmatrix} \qquad \underline{W}_{opt} = \begin{bmatrix} W_{1opt} \\ W_{2opt} \\ \cdot \\ \cdot \\ W_{Nopt} \end{bmatrix} \qquad (6.5)$$

where $W_{\ell opt} = W_\ell$ that satisfies (6.4) $\ell = 1,2,\ldots,N$.
Similarly define the matrix

$$C = <\underline{X}(t)\underline{X}^t(t)> = [<X_\ell(t)X_m(t)>] \qquad (6.6)$$

with $\underline{X}(t) = \begin{bmatrix} X_1(t) \\ \cdot \\ \cdot \\ X_N(t) \end{bmatrix}$ and $\underline{X}^t$ denoting the transpose of $\underline{X}$. The

solution of (6.4) then becomes

$$\underline{W}_{opt} = C^{-1}\underline{S} \qquad (6.7)$$

Equation (6.7) gives the weights that minimize ξ. These are
expressed in terms of the covariance matrix C that depends on
the statistics (second order) of the N stochastic inputs and
the column matrix (vector) $\underline{S}$ that involves the cross correlations
between the inputs and the desired response R(t). Thus, to find
the required weights one must know the covariance matrix and the
statistical dependence of R(t) on $X_\ell(t)$. In many applications
the covariance matrix of $X_\ell(t)$'s would not be known and the ex-
plicit inversion as indicated by (6.7) could therefore not be
carried out. An adaptive array carries this inversion process
out automatically by sensing the error $\varepsilon(t)$ and feeding it back
to adjust the weights until ξ is minimized, i.e., (6.7) is
satisfied. A possible closed loop system (adaptive filter) for
achieving this is shown in Figure 6.2. Here the symbols $\otimes$ and
$\triangle$ denote a multiplier and integrator, respectively.

Consider a typical loop. Performing the indicated opera-
tions yields

$$2K\, X_\ell(t)\, \varepsilon(t) = \frac{dW_\ell(t)}{dt}$$

Since $\varepsilon(t) = R(t) - \sum_{\ell=1}^{N} W_\ell(t)\, X_\ell(t)$, one finds

$$\frac{dW_\ell(t)}{dt} = -2K\, X_\ell(t) \sum_{m=1}^{N} W_m(t)\, X_m(t) + 2K\, R(t)\, X_\ell(t)$$

or, in matrix form

$$\frac{dW(t)}{dt} = -\, 2K\, \underline{X}(t)\underline{X}^t(t)\, \underline{W}(t) + 2K\, R(t)\, \underline{X}(t) \tag{6.3}$$
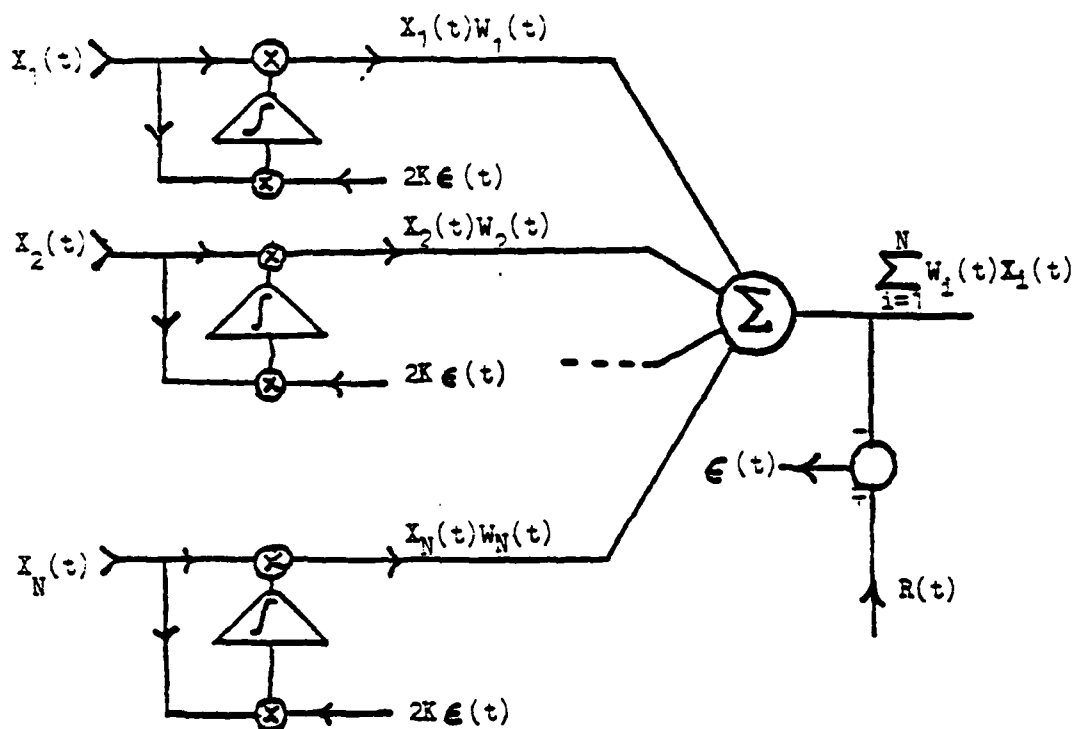
Figure 6-2 Widrow's Algorithm

This is a stochastic differential equation for the vector $\underline{W}(t)$.
It can be solved rigorously only if the detailed statistics of
$\underline{X}(t)$ and $\underline{R}(t)$ are known. However, if one assumes that the loop
response time is much longer than the decorrelation time
of $X(t)$ (in other words, the $X_\ell(t)$ vary much more rapidly than
the time constants of the adaptive process of adjusting the
weights), then one has

$$<\underline{X}(t) \ \underline{X}^t(t) \ \underline{W}(t)> \cong <\underline{X}(t) \ \underline{X}^t(t)><\underline{W}(t)> \quad , \tag{6.9}$$

i.e., $\underline{W}(t)$ remains essentially constant during the averaging
operation.

Taking averages of both sides of (6.8) yields

$$\frac{d<\underline{W}(t)>}{dt} = -2K<\underline{X}(t) \ \underline{X}^t(t)><\underline{W}(t)> + 2K<R(t) \ \underline{X}(t)> \tag{6.10}$$

If the convariance matrix $C = <\underline{X}(t) \ \underline{X}^t(t)>$ is positive definite
then (6.10) is stable and in the steady state (i.e., $t\to\infty$)

$$\frac{d<\underline{W}(t)>}{dt} \to 0$$

Thus, one finds that as $t\to\infty$

$$-2K<X(t) \ \underline{X}^t(t)><\underline{W}(t)> + 2K<R(t) \ X(t)>\to 0,$$

or

$$\lim_{t\to\infty} <\underline{W}(t)> \to W_{opt},$$

as given by (6.7). Thus, subject to the requirement that C be
positive definite and that $K>0$, the adaptive filter yields (6.7)
as the steady state solution. The settling time (i.e., the time

- 122 -

to reach steady state) is determined by K and the eigenvalues of
C. It will be convenient to examine the solution of (6.10) with
the aid of Laplace transforms. Let $\underline{w}(s)$ be the Laplace transform
of W(t). Then (6.10) is equivalent to

$$-W(0^+) + s\underline{w}(s) + 2KC\ \underline{w}(s) = \frac{2K}{s}\ \underline{S} \qquad (6.11)$$

Let

$\underline{w}'(s) = T\underline{w}(s)$, where T diagonalizes C and $TT^t = I$. Then
(6.11) may be written

$$-\underline{w}(0^+) + ST\underline{w}'(s) + 2KT^t\ \Lambda\ T\underline{w}(s) = \frac{2K}{s}\underline{S}$$

or

$$-T\underline{w}(0^+) + S\underline{w}'(s) + 2K\ \Lambda\ \underline{w}'(s) = 2K\ T\underline{S}/s \qquad (6.12)$$

where $\Lambda = \text{diag.}\ (\lambda_1,\lambda_2\ldots\lambda_N)$.

Solving (6.12) for $\underline{w}'(s)$,

$$\underline{w}'(s) = (s + 2K\ \Lambda)^{-1}\ T\underline{w}(0^+) + \frac{2K}{s}(s + 2K\ \Lambda)^{-1}\ T\underline{S} \qquad (6.13)$$

Inverting the transform in (6.13)

$$W'(t) = \begin{bmatrix} \ddots & & 0 \\ & e^{-2K\lambda_\ell t} & \\ & & \ddots \\ 0 & & \ddots \end{bmatrix} T\underline{w}(0^+) + \begin{bmatrix} \ddots & & 0 \\ & \frac{1}{\lambda_\ell} & \\ & & \ddots \\ 0 & & \ddots \end{bmatrix} T\underline{S} - \begin{bmatrix} \ddots & & 0 \\ & \frac{1}{\lambda_\ell}e^{-2K\lambda_\ell t} & \\ & & \ddots \\ 0 & & \ddots \end{bmatrix} T\underline{S}$$

To find $\underline{W}(t)$, the above is multiplied by T^t from the left to yield

$$\underline{W}(t) = T^t \begin{bmatrix} \ddots & & 0 \\ & e^{-2kt\lambda_\ell} & \\ & & \ddots \\ 0 & & \ddots \end{bmatrix} T\underline{w}(0^+) - T^t \begin{bmatrix} \ddots & & 0 \\ & \frac{e^{-2Kt\lambda_\ell}}{\lambda_\ell} & \\ & & \ddots \\ 0 & & \ddots \end{bmatrix} T\underline{S} + C^{-1}\underline{S} \qquad (6.14)$$
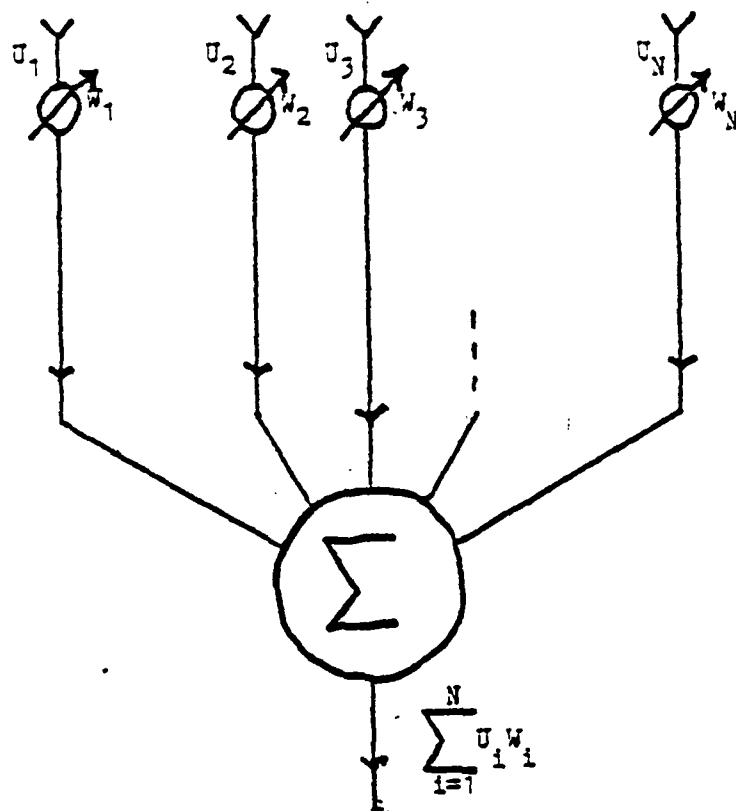
- 123 -

Figure 6-3 Adaptive Array with Complex Inputs

Evidently, the time constant of the system is given by

$$\tau = \frac{1}{2K\lambda_{min}} \qquad (6.15)$$

where λ_{min} is the smallest eigenvalue of C.

The preceding discussion followed closely the adaptation process described by Widrow.[1] The implementation is sometimes referred to as the LMS algorithm. A somewhat different phrasing and implementation of the adaptive process is due to Applebaum.[2] It is most appropriate for application to array antennas. First, consider an N-element array with outputs at the array ports u_1, $u_2, \ldots u_N$. The outputs at the array elements are complex time functions corresponding to the envelope in the analytic signal representation of the composite signal (including modulation) and noise. When no noise (generalized interference which may be ambient noise together with jamming) is present

$$u_k = \alpha \, S_k \qquad (6.16)$$

where S_k are weighted element patterns as they usually appear in the array factor (e.g., $S_k = m_k \, e^{j\underline{k}\cdot\underline{D}_k}$) while $\alpha = \alpha(t)$ is the information bearing signal. When only noise (interference) is present at the array output ports

$$u_k \equiv n_k(t) \qquad (6.17)$$

which may be composed of the ambient noise $n_{o,k}(t)$ and jammer noise $n_{J,k}(t)$, viz.

$$u_k = n_{o,k}(t) + n_{J,k}(t) \qquad (6.18)$$

Suppose that in the "quiescent" environment $(n_{J,k}(t) = 0)$ the weights for optimum signal reception are W_{kq}, or

$$\begin{bmatrix} W_{1q} \\ W_{2q} \\ \cdot \\ \cdot \\ \cdot \end{bmatrix} = \underline{W}_q$$

Then upon summation the desired signal is $\underline{W}_q \underline{t}$, i.e., $\underline{t} = \underline{s}$ is the vector that determines the correct array illumination under quiescent conditions. When noise only (interference) is present, the power output after array output summation is

$$< |\underline{n}^t \underline{W}|^2 > = \underline{W}^+ < \underline{n}^* \underline{n}^t > \underline{W} = \underline{W}^+ C \underline{W} \qquad (6.19)$$

where C is the covariance matrix of interference, $\underline{A}^+$ is the conjugate to transpose of A, and p^* is the complex conjugate of p. If the quiescent noise and jammer noise are uncorrelated

$$C = C_o + C_J$$

The signal to "noise" ratio can now be defined as

$$\frac{S}{N} = \frac{< |\alpha|^2 > \cdot |\underline{W}_q^t \underline{t}|^2}{\underline{W}^+ C \underline{W}} \qquad (6.20)$$

and one would like to choose $\underline{W}$ so that $\frac{S}{N}$ is maximum. Writing

$$|\underline{W}^t \underline{t}|^2 = \underline{W}^+ \underline{t}^* \underline{W}^t \underline{t} = \underline{W}^+ (\underline{t}^* \underline{t}^t) \underline{W}$$

$$\frac{S}{N} = <|\alpha|^2> \cdot \frac{\underline{W}^+ (\underline{t}^* \underline{t}^t) \underline{W}}{\underline{W}^+ C \underline{W}}$$

One readily finds that the preceding is maximized by

$$\underline{W} = W_{opt} = C^{-1} \underline{t}^* \qquad\qquad (6.21)$$

The adaptive implementation that realizes this optimum is shown in Figure 6.4 (for the kth element).

Let $RC = \tau$ be the time constant of the integrator. Then

$$-u_k^* \sum_{\ell=1}^{N} u_\ell W_\ell(t) + t_k^* = \frac{1}{G} (\tau \frac{dW_k}{dt} + W_k) \ .$$

Taking averages and reverting to matrices

$$- C \underline{W}(t) + \underline{t}^* = \frac{1}{G} (\tau \frac{dW_k}{dt} + \underline{W}(t))$$

or

$$\tau \frac{d\underline{W}}{dt} + \underline{W} = -G \, C \, \underline{W} + G \underline{t}^* \qquad\qquad (6.22)$$

The steady state solution is obtained by setting the time derivative to zero. Hence
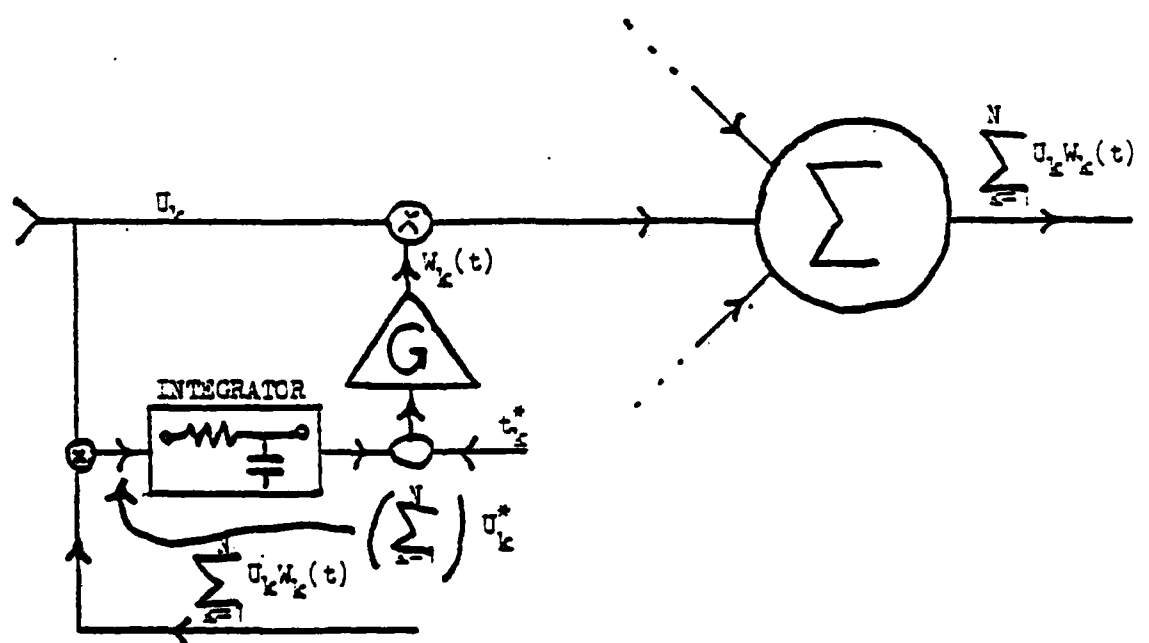
Figure 6-4 Implementation of Applebaum Algorithm

$$[G + I \tfrac{1}{G}] \, \underline{W} = \underline{t}^{*}$$

$$\therefore \underline{W} = [C + \tfrac{1}{G} \, \underline{I}]^{-1} \, \underline{t}^{*} \qquad (6.23)$$

For sufficiently large gain, $\underline{W} \to \underline{W}_{opt}$. However, when G is finite, there is always a residual error.

Deterministic Reponse of Adaptive Arrays

In this section, the adaptive array will be examined from a deterministic point of view. Receiver noise will be neglected and any signals incident upon the array will be assumed to be deterministic. Although an oversimplification of the actual situation, the deterministic behavior of a device many times supplies the physical insight necessary to extract information from the more complex stochastic case.

Our discussion will be limited to the Widrow algorithm shown in Figure 6.2. There an M port array having N=M/2 in phase and N quadrature ports is considered. The transient behavior of the array will be examined in three situations of interest.

First, the behavior of the array will be examined when a step plane wave is normally incident upon the array. Since the Widrow algorithm that has been chosen contains no steering mechanism its optimum direction of reception is normal to the array. Thus once the initial transient has decayed, the array should be locked on the normally incident plane wave.

Proceeding with the analysis, the equation for the weights (6.8) is given by

$$\frac{d\underline{W}(t)}{dt} = -2K\underline{X}(t)\underline{X}^t(t)\ \underline{W}(t) + 2KR(t)\underline{X}(t) \qquad (6.24$$

$$\underline{W}(o) = \underline{W}_o$$

where the initial weights $\underline{W}_o$ have been arbitrarily chosen.
Assume the input signal is of the form

$$\underline{X}(t) = \underline{X}\ u(t) \quad , \quad \underline{X} = A\ \underline{\sigma} \qquad (6.25)$$

where A is the amplitude of the plane wave normally incident on
the array, $\underline{\sigma}$ is an array vector given by

$$\underline{\sigma} = \begin{pmatrix} \underline{1}_N \\ \underline{0}_N \end{pmatrix} \quad , \quad 2N = M \qquad (6.26)$$

where $\underline{1}_N$ and $\underline{0}_N$ are column vectors of 1's and 0's respectively
and $u(t)$ is the Heavside function. The array vector takes the
form of (6.26) since the incident plane wave is assumed to excited
only in-phase components. Finally the reference signal is chosen
as $R(t) = A$.

Using the above information in (6.24)

$$\frac{d\underline{W}(t)}{dt} = -2K\underline{X}\ \underline{X}^t\ \underline{W}(t) + 2KA\underline{X} \qquad t \geq 0 \qquad (6.27)$$

$$\underline{W}(o) = \underline{W}_o$$

This is simply a $2N^{th}$ order system of ordinary differential
equations (o.d.e.'s) with constant coefficients. A solution
can be written as

$$W(t) = 2KA \int_0^t \exp[2K\underline{X}\ \underline{X}^t(t'-t)]dt' \qquad (6.28$$

$$+ \exp[-2K\underline{X}\ \underline{X}^t t]\underline{W}_o \ ; \qquad t \geq 0$$

- 130 -

The expression for $\underline{W}(t)$ given above can be substantially simplified by making use of the following identity:

$$\exp a \; \underline{X} \; \underline{X}^t = I + (\exp a|\underline{X}|^2 - 1) \frac{\underline{X} \; \underline{X}^t}{|\underline{X}|^2} \qquad (6.29)$$

where $|\underline{X}|^2 = \underline{X}^t\underline{X}$, I is the identity matrix and a is a constant. This identity is obtained in Appendix A.

Using (6.29) in (6.28) with $|\underline{X}|^2 = A^2N$, we have

$$\underline{W}(t) = A(1-e^{-2KNA^2t})\frac{\underline{X}}{N} + \underline{W}_o \qquad (6.30)$$
$$+ (e^{-2KNA^2t}-1)\frac{\underline{X} \; \underline{X}^t}{N} \; , \; t \geq 0$$

Since an explicit expression for $\underline{W}(t)$ has been obtained the output of the adaptive array, $y(t)$ can be calculated

$$y(t) = \underline{X}^t\underline{W}(t)$$
$$y(t) = A(1-e^{-2KNA^2t}) + \underline{X}^t \cdot \underline{W}_o e^{-2KNA^2t} \qquad (6.31)$$

In the steady state, the desired output is,

$$\lim_{t \to \infty} y(t) = A$$

In fact, the transient decays with a time constant $t = 1/2KNA^2$. Thus, the larger the factor KNA^2 becomes the faster the array locks unto the desired signal. We note that if $\underline{W}_o = \underline{X}/N$ then $y(t) = A$ for $t \geq 0$. This essentially means the array was in steady

- 131 -

state at t=0 and no transient was needed to match the initial conditions to the steady state.

Next an analysis will be made of a step jammer obliquely incident on an array. We will assume that the array is in steady state before the jamming signal intercepts the antenna. Let

$$\underline{X}(t) = \underline{X} + \underline{X}_J u(t) \qquad (6.32)$$

where $\underline{X}$ is given in (6.25) and

$$\underline{X}_J = B\underline{\sigma}_J \quad , \quad \underline{\sigma}_J = \begin{pmatrix} \underline{\sigma}_{JC} \\ \\ \underline{\sigma}_{JS} \end{pmatrix} \qquad (6.32a)$$

and

$$\underline{\sigma}_{JC} + j\underline{\sigma}_{JS} = (e^{j\alpha_J}, e^{2j\alpha_J}, \ldots, e^{Nj\alpha_J})^t \quad , \quad \alpha_J = \frac{2\pi d}{\lambda}\sin\vartheta_J \quad (6.33)$$

Here d is the array element spacing, λ is the wavelength, ϑ_J is the angle of incidence measured from the normal to the array and B is amplitude of the jamming signal.

Using this in (6.24) with R(t)=A, we have

$$\frac{d\underline{W}(t)}{dt} = -2K\hat{\underline{X}}\,\hat{\underline{X}}^t\underline{W}(t) + 2KA\hat{\underline{X}} \qquad (6.34)$$

$$\underline{W}(o) = \frac{X}{N} \quad , \quad t \geq 0$$

where

$$\hat{\underline{X}} = \underline{X} + \underline{X}_J \qquad (6.35)$$

The initial setting for the weights $\underline{W}_o$ has been chosen so that the array will be in steady state (y=A) at t=0. When the methods used in the previous section are applied to this case,

we find

$$\underline{W}(t) = A(1-e^{-2K|\hat{\underline{X}}|^2 t}) \frac{\hat{\underline{X}}}{|\hat{\underline{X}}|^2}$$

$$+ [I + (e^{-2K|\hat{\underline{X}}|^2 t}-1) \frac{\hat{\underline{X}} \hat{\underline{X}}^t}{|\hat{\underline{X}}|^2} \frac{\underline{X}}{N} , \quad t \geq 0 \qquad (6.36)$$

A direct calculation gives the output of the array once the weight vector is known. One finds

$$y(t) = \hat{\underline{X}}^t \underline{W}(t)$$

$$y(t) = A + \frac{ABJ}{N} e^{-2K|\hat{\underline{X}}|^2 t} \qquad (6.37)$$

.where

$$J = \underline{\sigma}_J^t \underline{\sigma} = \underline{\sigma}^t \underline{\sigma}_J \qquad (6.38)$$

The decay constant τ is:

$$\tau = [2K|\hat{\underline{X}}|^2]^{-1} \qquad (6.39)$$

Since

$$|\hat{\underline{X}}|^2 = |\underline{X} + \underline{X}_J|^2 = |A\underline{\sigma} + B\underline{\sigma}_J|^2$$

$$= (A^2 + B^2)N + 2ABJ]^{-1}$$

we have

$$\tau = [(A^2 + B^2)N + 2ABJ]^{-1} \qquad (6.40)$$

For large B, we find

$$\tau = \frac{1}{NB^2} \qquad (6.41)$$

Thus we see from (6.37) and (6.41) that although the transient has a higher amplitude, the larger the jamming signal be-

comes, the decay rate also becomes much more rapid. For large N, the behavior is more complex since J depends on N and can be negative.

Before preceding on to the periodic case, we shall point out an important weakness of the deterministic approach to the problem. We have seen from (6.37) that as t becomes large the output of the array approaches the required steady state, however, an examination shows that no null has developed in the array in the direction of the jamming signal.

This anomalous behavior in the deterministic case leads us to the conclusion that the adaptive processor has taken advantage of the special properties of the input signals to the array to cancel out the jammer signal. In general, one would suspect that in order for the array to cancel out a random jamming signal, a null would have to be formed in the direction of the jamming signal; thus, we are led to the conclusion that certain characteristics of the array could be different in the random case. This is indeed the case as we shall show later.

Before treating the random case, we shall consider the deterministic steady state response of the array to a periodic jammer. Although the method suffers from the previously mentioned weakness, the work serves to indicate methodology and gives insight into certain aspects of the arrays behavior in the presence of periodic signals.

To proceed, we take $\underline{X}(t)$ to be

$$\underline{X}(t) = \underline{X} + \underline{X}_J \ S(t) \tag{6.41}$$

where $S(t)$ is a periodic function of period T consisting of pulses of unit height and duration τ as shown in Figure 6.5.

We now look for solutions to (6.24) that are periodic, i.e.,

$$W(t+T) = W(t)$$

Our technique is as follows. First, assume $\underline{W}(t)$ is known at $t=0$, i.e., $\underline{W}(o) = \underline{W}_o$. Now solve (6.24) in the region of $0 \leq t < \tau$. Next, use the continuity of $\underline{W}$ at $t=\tau$ to establish initial conditions for the region $\tau < t < T$. Now solve (6.24) in this region and set $\underline{W}(T) = \underline{W}_o$. This gives an algebraic system of equations for $\underline{W}_o$. Upon execution of the foregoing, we find

$$\underline{W}(t) = A(1-e^{-2K|\hat{\underline{X}}|^2 t}) \frac{\hat{\underline{X}}}{|\underline{X}|^2} + [I+(e^{-2K|\hat{\underline{X}}|^2 t}-1) \frac{\hat{\underline{X}}\,\hat{\underline{X}}^t}{|\hat{\underline{X}}|^2}] \, \underline{W}_o \quad (6.43)$$

$$0 \leq t \leq \tau$$

and

$$\underline{W}(t) = A(1-e^{-2KNA^2(t-\tau)})\frac{\underline{X}}{N} + [I + (e^{-2KNA^2(t-\tau)}-1)\frac{\underline{X}\,\underline{X}^t}{N}]\underline{W}(\tau) \quad (6.44)$$

$$\tau \leq t \leq T$$

The equation for $\underline{W}_o$ is given by

$$\underline{W}_o = AP_o \frac{\underline{X}}{N} + [I-P_o \frac{\underline{X}\,\underline{X}^t}{N}][\underline{W}_o + P_1\frac{\hat{\underline{X}}}{|\underline{X}|^2}(A - \hat{\underline{X}}^t\underline{W}_o)] \quad (6.45)$$

where

$$P_o = 1-e^{-2KNA^2(T-\tau)}$$

$$P_1 = 1-e^{-2K|\hat{\underline{X}}|^2 \tau} \quad .$$

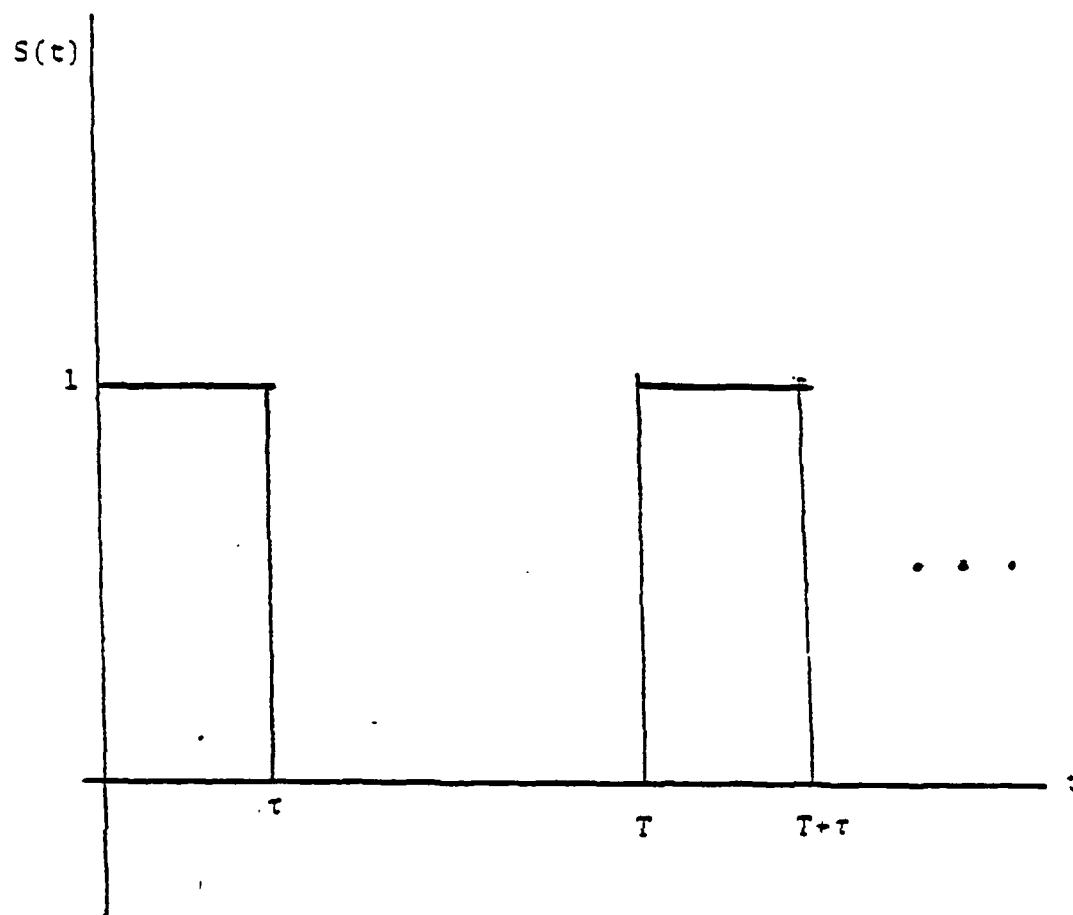Figure 6-5   Periodic Jammer Waveform

For arbitrary N, the system of equations for $\underline{W}_o$ is difficult to solve explicitly. Since only $y(t)$ is required, the complete solution to (6.45) is not required but only the projection of $\underline{W}_o$ in the $\underline{X}$ and $\hat{\underline{X}}$ directions, i.e., $\hat{\underline{X}}^t \underline{W}_o$ and $\underline{X}^t \underline{W}_o$. Let us define

$$Y_o^+ = \hat{\underline{X}}^t \underline{W}_o \tag{6.46}$$

$$Y_o^- = \underline{X}^t \underline{W}_o \tag{6.47}$$

where $Y_o^{\pm}$ are scalar quantities which determine $y(t)$. This can be seen from the fact that

$$y(t) = \begin{cases} \hat{\underline{X}}^t \underline{W}(t) & , \quad 0 \le t \le \tau \\ \\ \underline{X}^t \underline{W}(t) & , \quad \tau \le t < T \end{cases} \tag{6.48}$$

If (6.43) and (6.44) are used in (6.48), we obtain

$$y(t) = \begin{cases} 1 + (Y_o^+ - 1)e^{-2K|\hat{\underline{X}}|^2 t} & , \quad 0 < t < \tau \\ \\ 1 + (Y_o^- - 1)e^{+2KN^2A^2(T-t)} & , \quad \tau < t < T \end{cases} \tag{6.49}$$

Equations for $Y_o^{\pm}$ can be obtained by multiplying (6.45) from the left by $\hat{\underline{X}}^t$ and $\underline{X}^t$. We find

$$- \frac{P_o \hat{\underline{X}}^t \underline{X}}{N} Y_o^- + P(P_o \frac{\hat{\underline{X}}^t \underline{X} \underline{X}^t \hat{\underline{X}}}{N |\underline{X}|^2} - 1) Y_o^+ = g_1 \tag{6.50}$$

$$- P_o Y_o^- + P(P_o - 1)\frac{\underline{X}^t \hat{\underline{X}}}{|\underline{X}|^2} Y_o^+ = g_2 \tag{6.51}$$

where

$$g_1 = A[\frac{P_o \, P \, \hat{\underline{X}}^t \, \underline{X} \, \underline{X}^t \, \hat{\underline{X}}}{N \, |\hat{\underline{X}}|^2} - P - P_o \frac{\hat{\underline{X}}^t \, \underline{X}}{N}]$$

$$g_2 = A[\frac{P_o \, P \, \underline{X}^t \, \hat{\underline{X}}}{|\hat{\underline{X}}|^2} - P_o - P \frac{\underline{X}^t \, \hat{\underline{X}}}{|\hat{\underline{X}}|^2}]$$

The solution of this system (6.50-51) for $Y_o^{\pm}$ is straight forward and yields $Y_o^{\pm} = 1$. By using this result in (6.49), we find that

$$y(t) = 1 \quad , \qquad 0 \le \tau \le T \tag{6.52}$$

The physical interpretation of this result is that the adaptive array has had an infinite amount of time to adapt to the presence of a periodic jammer, and as in the case of the step jammer, it adapts perfectly. This is an important result. It indicates that in order to defeat an adaptive array, the jamming signal must be constantly changing in a random fashion.

Random Response of Adaptive Arrays

Before concluding this Chapter, we would like to present an interesting analysis of the approximate mean equation given in (6.10). Although, the transient character of this equation has been treated in the literature and reviewed in this report (6.11-6.15), many of the symmetry properties of the equation were not adequately utilized. An indication of this is illustrated by (6.15). Here the transient properties of the array are given in terms of the minimum eigenvalue, λ_{min},

however, no information about the behavior of λ_{min} with respect to the system parameters such as N, A, etc., is given.

We shall rectify the situation by utilizing the group properties of the system of equations of the weights $\underline{W}(t)$. It is shown in Appendix A that by proper parameterization of the problem, the 2N order system of equations can be replaced by a fifth order system for any N. This system is solved explicitly in the constant coefficient case, and a generalization of the identity used in the deterministic case is derived. We use this identity in this section to obtain our main result.

Although the method we employ has broad application, we use it just to analyze the case of a broad spectrum step jammer.

The analysis proceeds as follows. Consider (6.10) with $R(t) = A$:

$$\frac{d \, <\underline{W}(t)>}{dt} = -2K <\underline{X}(t) \, \underline{X}^t(t)><\underline{W}(t)> + 2KA <\underline{X}(t)> \qquad (6.53)$$

$$\underline{W}(o) = \underline{W}_o$$

where

$$\underline{X}(t) = A\underline{\sigma} + N_J(t) \, \underline{\sigma}_J \qquad (6.54)$$

Here $\underline{\sigma}$ and $\underline{\sigma}_J$ are array factors as defined in (6.26) and (6.32a), respectively and $N_J(t)$ is a zero mean stationary Gaussian processes with spectral density N_{oJ}.

Using this information, we have

$$<\underline{X}(t) \, \underline{X}^t(t)> = A^2 \, \underline{\sigma} \, \underline{\sigma}^t + N_{oJ} \, \underline{\sigma}_J \, \underline{\sigma}_J^t \qquad (6.55)$$

and

$$\langle \underline{X}(t) \rangle = A\underline{\sigma} \qquad (6.56)$$

Putting (6.55) and (6.56) into (6.53) an equation with constant coefficients results. The solution to this equation is

$$\langle \underline{W}(t) \rangle = 2KA \int_0^t e^{Q(t'-t)} \, dt' \, \underline{\sigma} + e^{-Qt}\underline{W}_0 \qquad (6.57)$$

where

$$Q = 2K[A^2 \, \underline{\sigma} \, \underline{\sigma}^t + N_{0J} \, \underline{\sigma}_J \, \underline{\sigma}_J^t] \qquad (6.58)$$

The expression in (6.57) can be substantially simplified by using the identity given in Appendix A. We have

$$e^{Qt} = f_0(\underline{z}) + f_+(\underline{z})e^{\lambda^+ t} + f_-(\underline{z})e^{\lambda^- t} \qquad (6.59)$$

where

$$\lambda^{\pm} = \frac{(a+b)N \pm \sqrt{(a-b)^2 N^2 + 4abJ^2}}{2} \qquad (6.60)$$

with

$$a = 2KA^2 \quad , \quad b = 2KN_{0J} \qquad (6.61)$$

The vector $\underline{z}$ appearing in (6.59) consists of five components. Each component is a matrix $z_0, \ldots, z_4$. These matrices form a complete set of generators for a group under matrix multiplication. It is the realization that expQt is also a member of this group that leads to the identity (6.59). The explicit formula for z_i, $i=0,\ldots 4$ and the functions f_0 and $f_{\pm}$ are given in Appendix A.

The important property of (6.59) to notice is that $\exp(Qt)$ has been represented in terms of three scalar functions of

time whose exponential decay rates are given explicitly in terms of the parameters of the system. Using (6.59) in (6.57) and integrating, we obtain:

$$
\langle W(t) \rangle = 2KA \left[\frac{1 - e^{-\lambda^+ t}}{\lambda^+} f_+(\underline{z}) \cdot \underline{\sigma} \right.
$$

$$
\left. + \frac{1 - e^{-\lambda^- t}}{\lambda^-} f_-(\underline{z}) \cdot \underline{\sigma} \right]
$$

(6.62)

where we have used the fact that $f_o(\underline{z}) \cdot \underline{\sigma} = 0$.

Now that we have the exact solution to the approximate mean equation, we will examine its behavior in the steady state. Let

$$
\langle \underline{W}_\infty \rangle = \lim_{t \to \infty} \langle \underline{W}(t) \rangle
$$

(6.63)

By letting t approach infinity in (6.62), we find

$$
\langle \underline{W}_\infty \rangle = 2KA \left[\frac{f_+(\underline{z})}{\lambda^+} + \frac{f_-(\underline{z})}{\lambda^-} \right] \underline{\sigma}
$$

(6.64)

Using the expressions for f_o and $f_\pm$ given in Appendix A to simplify (6.64), we obtain

$$
\langle \underline{W}_\infty \rangle = \frac{(N\underline{\sigma} - J\underline{\sigma}_J)}{(N^2 - J^2)}
$$

(6.65)

Since the output of the adaptive array is given by $y(t) = \underline{X}^t(t) \underline{W}(t)$, we have for $t \to \infty$:

$$
\langle y_\infty \rangle = \lim_{y \to \infty} \langle y(\tau) \rangle = A \underline{\sigma}^t \langle \underline{W}_\infty \rangle
$$

(6.66)

$$
\langle y_\infty \rangle = \frac{A(N \underline{\sigma}^t \underline{\sigma} - J \underline{\sigma}^t \underline{\sigma}_J)}{N^2 - J^2}
$$

$$\langle y_\infty \rangle = A \qquad (6.67)$$

Thus we see that the jamming signal has been completely cancelled out in the steady state. One may recall that this is exactly the result obtained in the deterministic case.

Next, we would like to determine if the cancellation of the jamming signal was due to a null in the phased array in the direction of the jammer. To do this, we define the array amplitude directivity factor $F(\theta,t)$ by

$$F(\theta,t) = \underline{\sigma}_\theta^t \; \langle \underline{W}(t) \rangle \qquad (6.68)$$

where

$$\underline{\sigma}_\theta = \begin{pmatrix} \underline{\sigma}_{\theta C} \\ \\ \underline{\sigma}_{\theta S} \end{pmatrix} \qquad (6.69)$$

and

$$\underline{\sigma}_{\theta C} + j \, \underline{\sigma}_{\theta S} = (e^{j\alpha}, e^{2j\alpha}, \ldots, \tau^{Nj\alpha}) \qquad (6.70)$$

$$\alpha = \frac{2\pi d}{\lambda} \sin\theta$$

We note that the directivity factor is a function of time since as the weights change, the radiation pattern of the antenna changes.

We would now like to compare $F(o)$ to $F(\theta_J)$ for $t \to \infty$. Let

$$F_\infty(\theta) = \lim_{t \to \infty} F(\theta,\tau) = \underline{\sigma}_\theta^t \langle \underline{W}_\infty \rangle \qquad (6.71)$$

and then, using (6.65) we have

$$F(o) = 1$$

and

$$F(\theta_J) = 0 \quad .$$

Thus we conclude, that an infinitely deep null has been de-
veloped in the direction of the jamming signal. Thus we
see that in the random case the array factor is responsible
for nulling out the jamming signal rather then the signal can-
cellation that occurred in the deterministic case. The array
behavior in the random case requires just one set of steady
state weights to cancel out any jamming signal entering at
$\theta = \theta_J$ while the array behavior in the deterministic case re-
quires a different set of weights for each different jamming
waveform at $\theta = \theta_J$.

Before concluding this section we would like to discuss
the settling time of the array in the random case. An examin-
ation of (6.62) shows that there are two decay factors λ^+ and
λ^- given in (6.60). This is in contrast to the step jammer
considered in the deterministic case. In that case only one
decay factor appeared.

The time constant of the array to reach steady state is
given by

$$\tau_{sJ} = \frac{1}{\min(\lambda^+, \lambda^-)} \tag{6.72}$$

where

$$\min(\lambda^+, \lambda^-) = \begin{cases} \lambda^+, & \lambda^- > \lambda^+ \\ \\ \lambda^-, & \lambda^+ > \lambda^- \end{cases}$$

- 143 -

For large jamming amplitudes b>>a or $N_{oJ}>>A^2$, expression (6.60) can be substantially simplified. We find

$$\lambda^+ \approx bN$$

and

$$\lambda^- \approx \frac{a}{N} (N^2 - J^2) > 0$$

This shows that the λ^+ is the counterpart to the decay factor found in the deterministic case (6.41) while λ^- represents a new effect. When b becomes large it is clear that for fixed N, λ^- becomes the slowest decay constant and thus determines the transient time of the array.

It is generally believed that the larger the jamming signal, the faster the array will lock in. It is clear that the presence of λ^- will tend to mask the above effect for very large b since the array settling time will depend only on λ^- and this parameter is independent of b. Thus there appears to be a saturation effect. As b is increased, the transient time decreases until approximately $\lambda^+ > \lambda^-$ and then the transient time remains relatively fixed.

We point out that the above results are of a preliminary nature. Only the approximate equation for the mean has been solved, not the exact mean equation. In addition, no check of the variance has been made which is essential. Work is continuing on these topics.

The application of the techniques used in this section, to other types of jammers is certainly possible. In particular, the investigation of a periodic wide spectrum jammer could be treated by these methods. Although the problem has not been

attacked formally, it is our conjecture that the steady state array factor would have a perfect null in the direction of the jammer. This would indicate that to effectively jam an array the use of aperiodic modulation of the wide spectrum jamming signal would be more effective than a periodic modulation.

Conclusion

We have shown that the methods are at hand for assessing the vulnerability of adaptive null steering antenna arrays to broadside pulse, step CW, periodic and random jammers. A remarkable result is that for periodic jammers and for noise jammers the steady state null can be perfect (assuming a perfect reference). This leads us to the conclusion that null steering antennas would be most vulnerable to blinking, non periodic, wide spectrum jammers. For this situation it is important to understand the parameters that determine the array time constants. This we have arrived at explicitly for both deterministic and random signals. A further interesting aspect of the analysis is that by exploiting the symmetries in the array equations we are able to show that the array settling time constants are determined by three scalar functions of time whose exponential settling time are given explicitly in terms of the parameters of the system. This is true for any size array. This result will be of enormous benefit in the design and implementation of adaptive arrays. Adaptive arrays are the only ECCM technique which allows substantial anti-jam performance without bandwidth expansion.

References to Chapter 6.

1. B. Widrow, P. E. Mantey, L. J. Griffiths, and B. B. Goode, "Adaptive Antenna Systems," Proc. IEEE, $\underline{55}$, pp. 2143-2159, Dec., 1967.

2. S. P. Applebaum, "Adaptive Arrays," Syracuse University Res. Corp., Rep. SPL TR 66-1, Aug. 1966.

3. S. P. Applebaum, "Adaptive Arrays," IEEE, AP $\underline{24}$, pp. 585-598, Sept. 1976.

4. L. J. Griffins, "A Simple Adaptive Algorithm for Real Time Processing in Antenna Arrays," Proc. IEEE, $\underline{57}$, pp. 1696-1704, Oct. 1969.

5. O. L. Frost, III, "An Algorithm for Linearly Constrained Adaptive Array Processing," Proc. IEEE, $\underline{60}$, pp. 926-935, Aug. 1972.

6. L. E. Brennan, E. L. Pugh and I. S. Reed, "Control-loop Noise in Adaptive Array Antennas," IEEE, AES, I, pp. 254-262, Mar. 1971.

7. L. E. Brennan and I. S. Reed, "Effect of Envelope Limiting in Adaptive Array Control Loops," IEEE AES, I, pp. 698-700, July 1971.

8. L. E. Brennan and I. S. Reed, "Convergence Rate in Adaptive Arrays," Technology Service Corp. Report - PD-A177-3, Oct. 1977.

9. Compton, R. T., "An Improved Feedback Loop for Adaptive Arrays," To appear in Proc. IEEE.

Appendix to Chapter 6.

In this Appendix we will derive an identity that will be useful for simplifying expressions of the form $\exp\{(a\underline{\sigma}\,\underline{\sigma}^t + b\underline{\sigma}_J\underline{\sigma}_J^t)t\}$ where $\underline{\sigma}$ and $\underline{\sigma}_J$ have been defined in (6.26) and (6.32a), respectively and a and b are arbitrary constants.

We start by defining the matrices z_i, $i=0,1,\ldots,4$. They are

$$z_0 = I \qquad\qquad z_3 = \underline{\sigma}_J\,\underline{\sigma}^t$$

$$z_1 = \underline{\sigma}\,\underline{\sigma}^t \qquad\qquad z_4 = \underline{\sigma}_J\,\underline{\sigma}_J^t$$

$$z_2 = \underline{\sigma}\,\underline{\sigma}_J^t \tag{A6-1}$$

Next we recognize that $\exp\{(az_0 + bz_4)t\}$ is the fundamental solution to

$$\frac{dT(t)}{dt} = [az_0 + bz_4]T(t) \tag{A6-2}$$

$$T(0) = I$$

Following this we expand T in terms of the z_i. We have

$$T(t) = \sum_{i=0}^{4} \alpha_i(t)\,z_i \tag{A6-3}$$

Using (A6-3) in (A6-2), we obtain the following system of equations for $\alpha(t)$:

$$\frac{d\underline{\alpha}(t)}{dt} = A\,\underline{\alpha}(t) \quad , \quad \underline{\alpha} = \begin{bmatrix} \alpha_0 \\ \vdots \\ \alpha_4 \end{bmatrix} \tag{A6-4}$$

$$\underline{\alpha}(0) = \alpha_0$$

where

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ a & aN & 0 & aJ & 0 \\ 0 & 0 & aN & 0 & aJ \\ 0 & bJ & 0 & bN & 0 \\ b & 0 & bJ & 0 & bN \end{bmatrix} \quad , \quad \alpha_0 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \qquad (A6-5)$$

The eigenvalues of A, i.e., and $J = \underline{\sigma}^t \, \underline{\sigma}_J$.

$$\det|A - I\,\lambda| = 0 \qquad (A6-6)$$

are: $\lambda = 0$ of multiplicity 1, and $\lambda^\pm$ each of multiplicity 2 where

$$\lambda^\pm = \frac{(a+b)\,N \pm \sqrt{(a-b)^2\,N^2 + 4\,abJ^2}}{2} \gtrless 0 \qquad (A6-7)$$

One should note that although λ^+ has a multiplicity of two no growing terms in t result since each of the two eigenvalues for λ^+ have independent subspaces. The same is true for λ^-.

The matrix A can be diagonalized by the similarity transformation T as follows:

$$AT = T\Lambda \qquad (A6-8)$$

where

$$\Lambda = \mathrm{diag}(0, \ \lambda^+, \ \lambda^+, \ \lambda^-, \lambda^-) \qquad (A6-9)$$

and T is a 5 x 5 consisting of the eigenvectors of A. If we define

$$\underline{\alpha}(t) = T \, \underline{e}(t) \qquad (A6-10)$$

then (A6-4) becomes

$$\frac{d\underline{e}}{dt} = \Lambda \underline{e} \tag{A6-11}$$

$$\underline{e}(0) = T^{-1} \underline{\alpha}_0$$

The desired identity can now be obtained: first, solve (A6-11) for $\underline{e}(t)$, find $\underline{\alpha}(t)$ from (A6-10) then use (A6-3).

The result is

$$e^{[az_0 + bz_4]t} = f_0(\underline{z}) + f_+(\underline{z})e^{\lambda^+ t} + f_-(\underline{z})e^{\lambda^- t} \tag{A6-12}$$

where

$$\underline{z} = (z_0, z_1, \ldots z_4)^t$$

and

$$f_0(\underline{z}) = [(J^2 - N^2) z_0 + Nz_1 - Jz_2 - Jz_3 + Nz_4]/(J^2 - N^2) \tag{A6-13}$$

$$f_+(\underline{z}) = [aJz_1 + (\lambda^+ - aN) z_3]/aJN \tag{A6-14}$$

$$f_-(z) = \{-\frac{J}{N}(aJ z_1 + (\lambda^- - aN) z_3) \tag{A6-15}$$

$$+ aJ z_2 + (\lambda^- - aN) z_4\} \cdot [aN(J^2 - N^2)]^{-1}$$

$\lambda^{\pm}$ are given by (A6-10) and $J = \underline{\sigma}^t \underline{\sigma}_J$.

The formula simplifies substantially when b=0. We have

$$e^{a\underline{\sigma} \, \underline{\sigma}^t \, t} = \frac{e^{aNt} - 1}{N} \underline{\sigma} \, \underline{\sigma}^t + I \tag{A6-16}$$

CHAPTER 7

Modulation and Spreading Techniques

In conjunction with the dynamic allocation of channel capacity in Chapter 3 of this report the use of an appropriate combination of modulation/spreading technique will ensure that the system will accrue the maximum advantage with respect to a jammer. This latter topic, namely which modulation/spreading performs the best when used over a channel with a fixed bandwidth constraint, was studied in depth in a companion study performed by S Consulting Services entitled "Adaptive Techniques in Multi-channel Transmission." References [7.1] and [7.2] document both the analyses and the results. Because these results were documented in the above references quite extensively, they will not be repeated here. Rather an overview of the key conclusions will be presented, specifically with respect to their relevance to an overall system employing the dynamic allocation of channel capacity algorithm.

Two basic combinations of modulation/spreading were considered, the first being a coherent direct sequence (DS) system using either BPSK, QPSK, or 16-ary QASK, and the second being a frequency hopped (FH) system employing noncoherent FSK. In particular, in the latter system, the MARK and SPACE channels were allowed to overlap (i.e., they were not necessarily orthogonal) in order to provide more slots in a given bandwidth over which to hop.

- 151 -

Consider the DS systems. These systems were analyzed in the presence of both tone jamming and broadband noise jamming. It was found that for tone jamming the QPSK modulation technique performed the best, while with noise jamming BPSK and QPSK performed identically, both of them outperforming 16-ary QASK.

Physically, what is happening is the following. Since a higher level alphabet (e.g., 16-ary QASK) has a signal constellation which in general has a denser location of signal vectors (under a constraint of either fixed average power or fixed peak power), the higher order system pays a definite penalty with respect to thermal noise. On the other hand, for a fixed information rate the bandwidth of a higher order system can be reduced; hence it has the advantage of a higher processing gain in a fixed overall bandwidth. Thus there are two opposing forces at work, and for the systems considered here, the former one was the dominant one. That is, the greater vulnerability of the 16-ary system to thermal noise outweighed the decrease vulnerability of the system to the jammer.

Figures 7.1 - 7.4 show plots of probability of symbol error for the QPSK system under different jamming threats. Figures 7.1 and 7.2 correspond to a tone jammer, and Figures 7.3 and 7.4 correspond to a noise jammer. One qualification has to be mentioned with respect to Figures 7.1 and 7.2. The tone jammer for those figures was located at the carrier frequency of the QPSK signal, and an entire period of the spreading code was contained in one data symbol. As described in [7.2], this
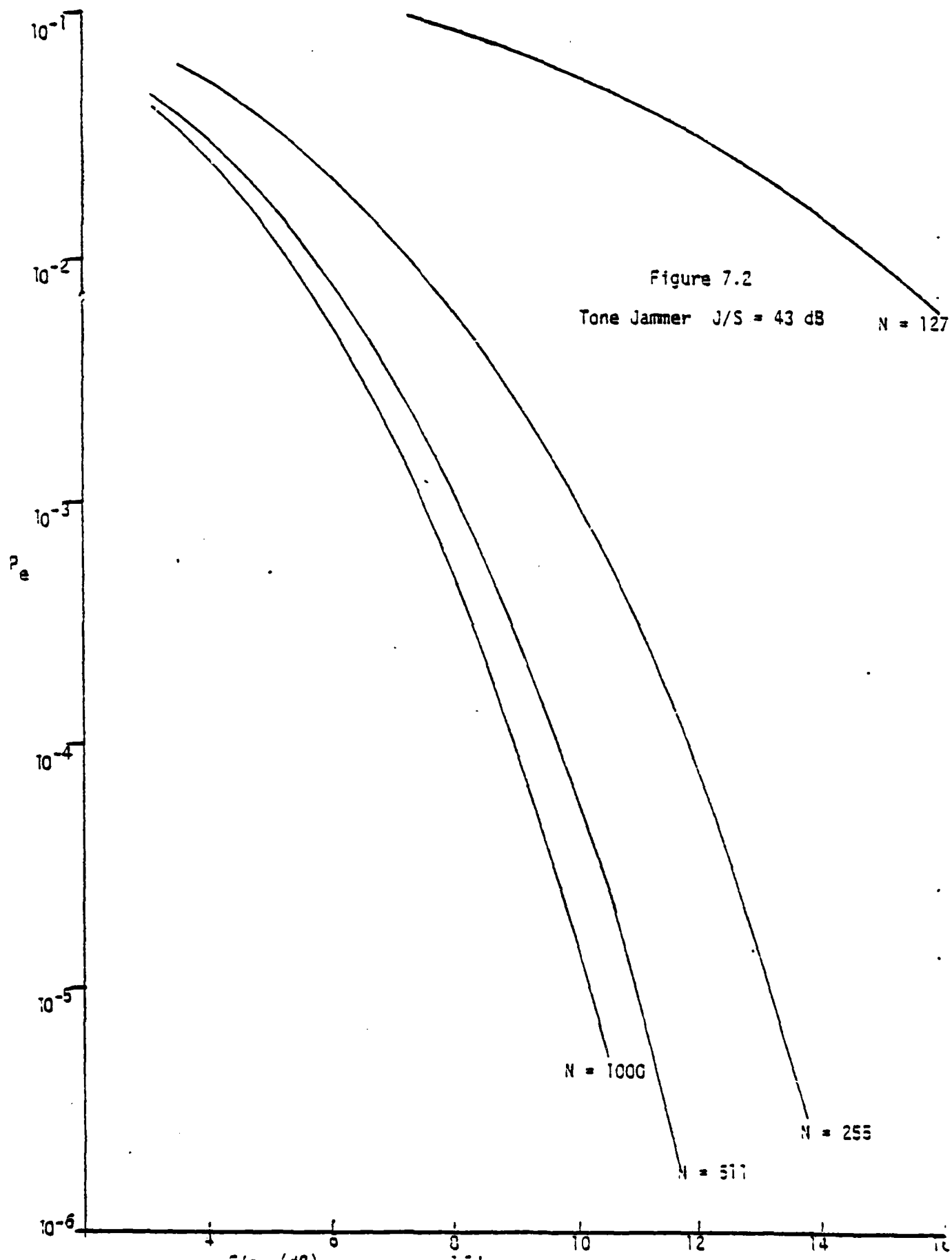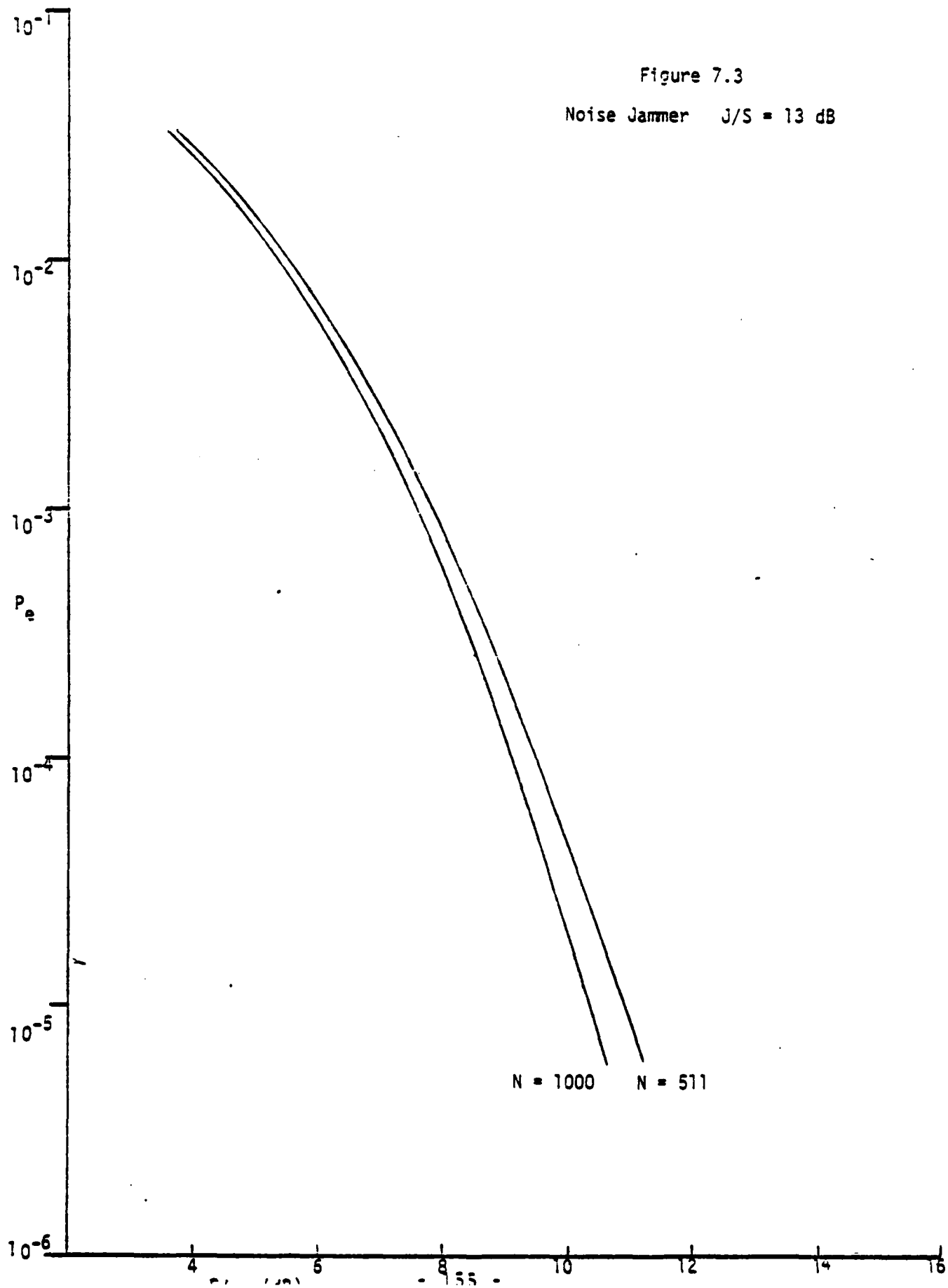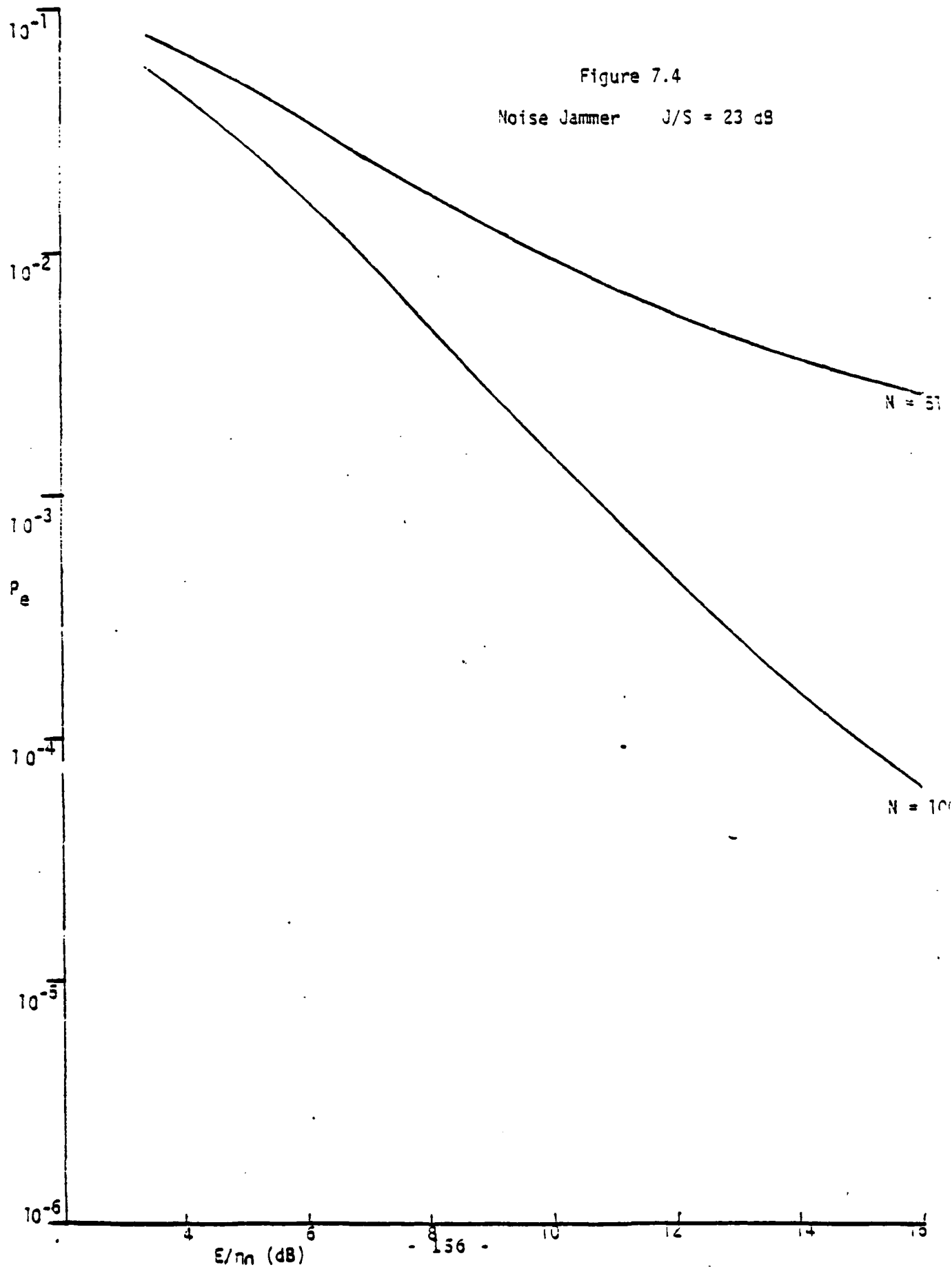
Figure 7.1

Tone Jammer J/S = 33 dB

P_e

E/η_0 (dB)

N=255

N=511

N=1000

N=127

- 153 -

Figure 7.2
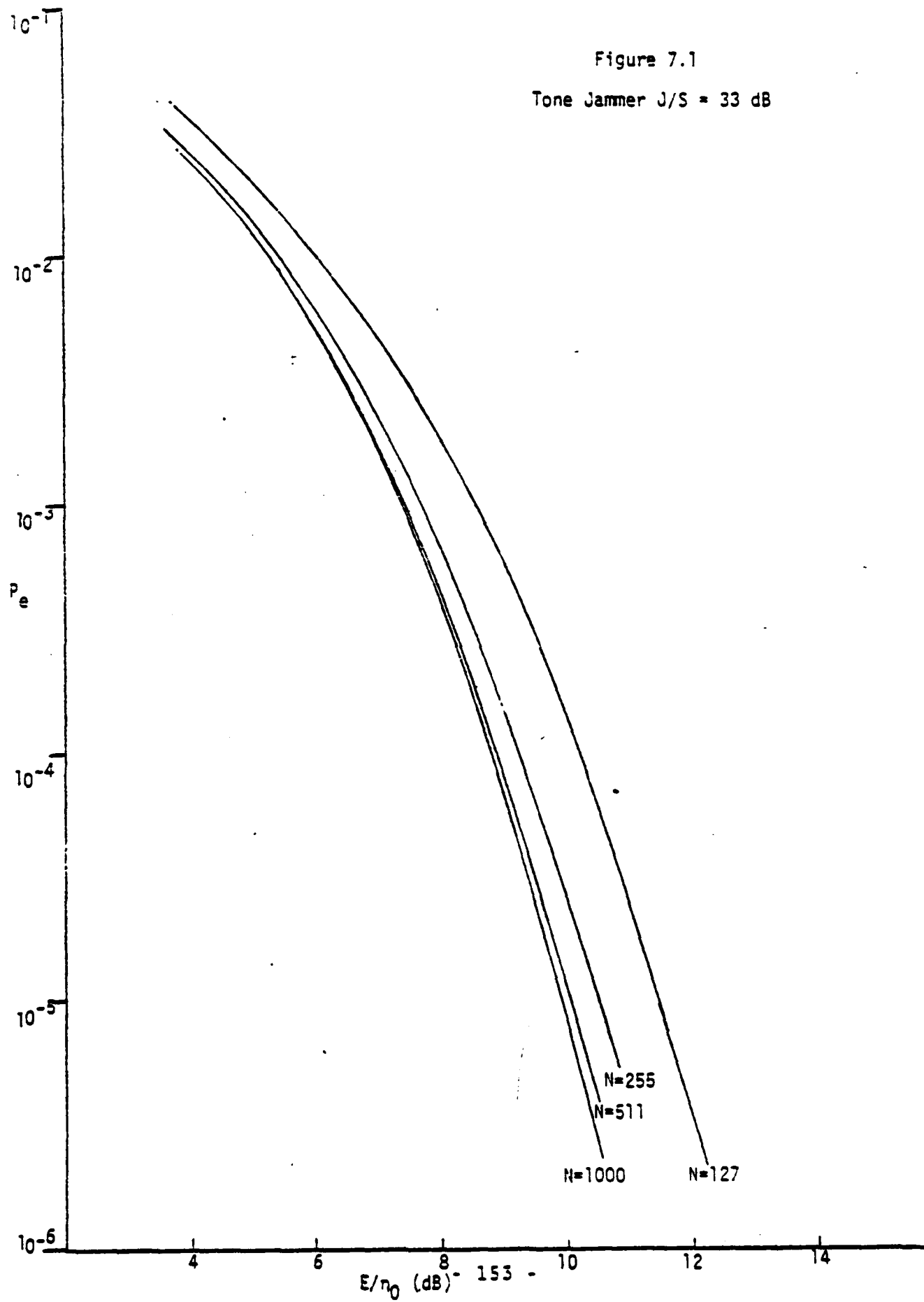
Tone Jammer   J/S = 43 dB

Figure 7.3

Noise Jammer   J/S = 13 dB

- 155 -

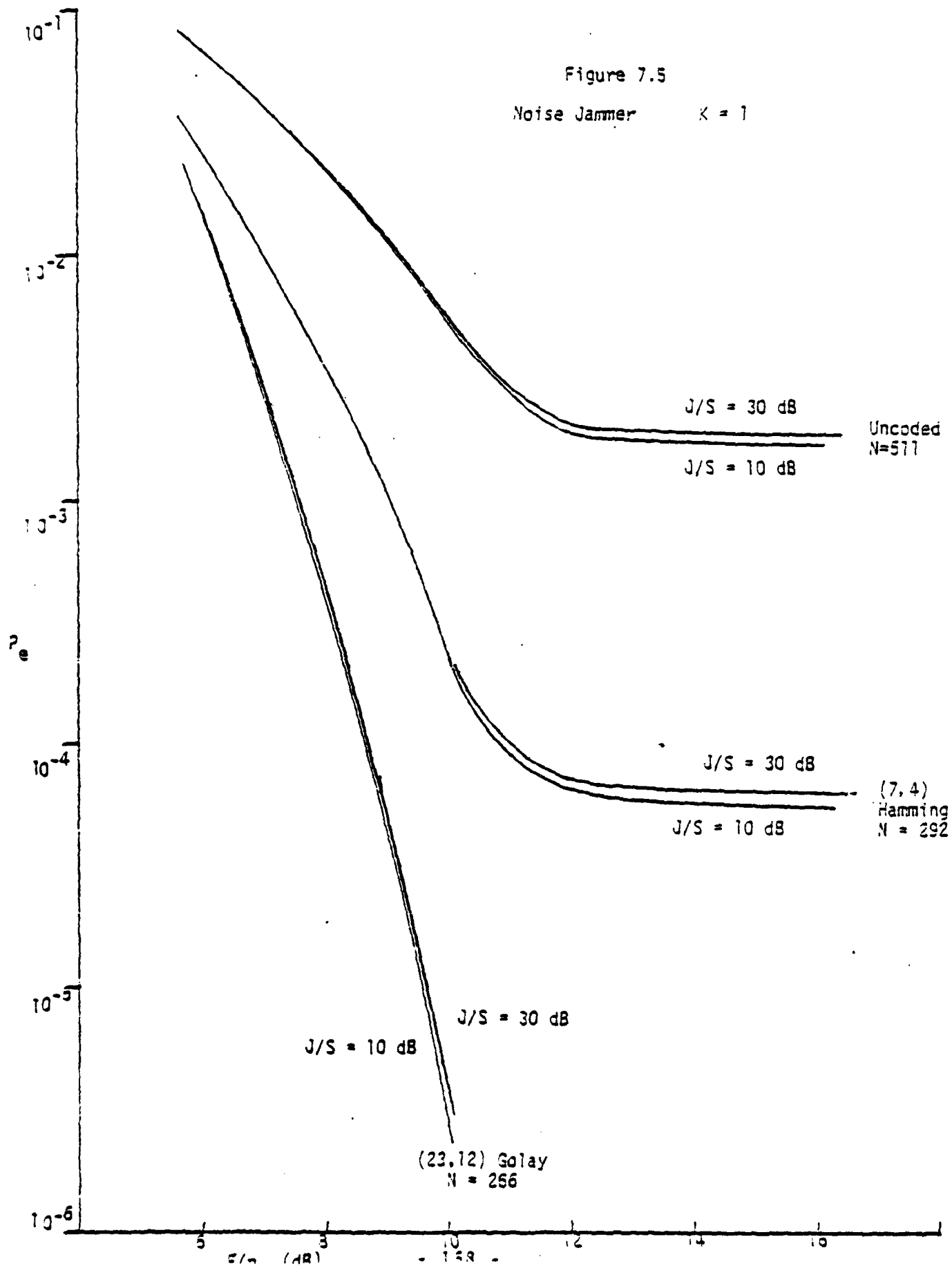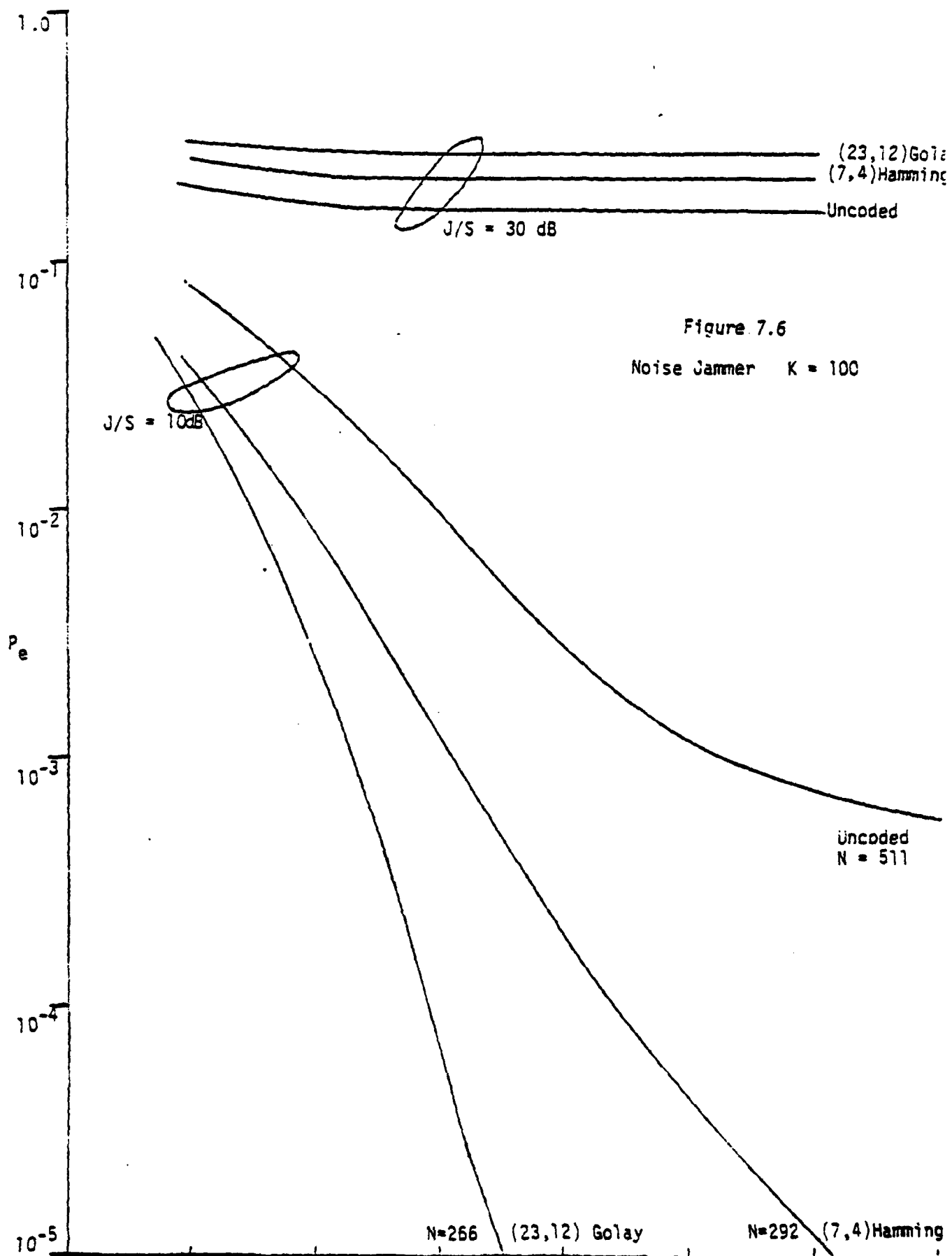Figure 7.4

Noise Jammer   J/S = 23 dB

combination, while allowing precise results to be calculated,
leads to an overly optimist'~ answer with respect to the situation
of say a tone jammer locat   roximately 1/T Hz away from
the carrier frequency where   s the symbol duration. In other
words, for this latter situation the ratios of jammer power-to-
signal power (J/S) for which the results of Figures 7.1 and 7.2
apply would be much smaller than what is indicated on the figures.

With respect to the FSK-FH systems, it was found that some
type of error-correction coding was needed in order for the system
to perform satisfactorily in the presence of a partial-band
jammer. Otherwise the jammer could put all its power in just a
single slot and cause an error rate of approximately 1/N, where
N is the number of slots over which the signal can hop, indepen-
dently of the input signal-to-noise ratio. Because of this, the
system was evaluated with a (7,4) Hamming cude and a (23, 12)
Golay code. As is well known ([7.3]), the former code corrects
all single errors and the latter code corrects all combinations
of three or fewer errors. Results of system performance for an
orthogonal FSK system are shown in Figures 7.5 and 7.6. When the
Golay encoded system was evaluated for both a 25% and a 50% overlap
between the MARK and SPACE channels (and hence a corresponding
increase in slots over which to hop) it was found there was very
little difference in performance from what is shown in Figures
7.5 and 7.6 ([7.2]).

It is now possible to put these results in perspective with
the results of Chapter 3. The question that was left unanswered
at the end of Chapter 3 was how much does one gain in the way of

Figure 7.5

Noise Jammer   K = 1

J/S = 30 dB

J/S = 10 dB

Uncoded
N=511

J/S = 30 dB

J/S = 10 dB

(7, 4)
Hamming
N = 292

J/S = 30 dB

J/S = 10 dB

(23,12) Golay
N = 256

P_e

E/n (dB)

- 158 -

Figure 7.6

Noise Jammer   K = 100

system improvement by, say, reducing the number channels by 50% and correspondingly increasing the processing gain by 50%. From the figures it is clear that, depending upon the processing gain E_b/N_o, and J/S, one could pick up anywhere from 0 dB to tens of dB's in improvement.

For example, from Figure 7.3, corresponding to J/S = 13 dB, if the probability of error of the system is 10^{-4}, going from 511 channels to 1000 channels results in an improvement of about 0.4 dB. From Figure 7.4 however, corresponding to J/S = 23 dB, the improvement at the same error rate is essentially unbounded, since the curve for N = 511 has started flattening out at around 2.6×10^{-3}. Therefore, while the actual improvement in system performance that one obtains by dynamically varying the number of channels in use depends upon the specific operating conditions of the channel, it is seen that for any system that is being highly stressed, the improvement can be enormous.

In summary then, using either coherent QPSK in a DS spread spectrum system, or noncoherent FSK in a FH system (in conjunction with an error-correcting code) can, when combined with a dynamic channel capacity allocation algorithm, result in significant improvement of performance of a system being severely jammed.

References

[7.1] "Adaptive Techniques in Multichannel Transmission,"
Interim Report for 9/30/78 - 1/31/79, prepared for U.S.
Army CORACOM by S Consulting Services.

[7.2] "Adaptive Techniques in Multichannel Transmission,"
Final Report.

[7.3] R. G. Gallager, <u>Information Theory & Reliable Communication</u>,
Wiley, 1968.

CHAPTER 8

Conclusions And Recommendations For Further Work

In this document we reported on the analysis of a variety
of techniques for reducing ECM vulnerability of tactical communi-
cation. The specific techniques chosen for study included:

1) The use of traffic control and channel capacity allocation
to permit certain classes of users to maintain communication under
severe jamming conditions.

2) A technique using extreme value theory for the purpose of
monitoring the bit error rate which can then be used in conjunction
with (1) to provide a completely adaptive and dynamic channel
allocation scheme that will respond directly to jamming threats.

3) The search for more bandwidth efficient digital modulation
schemes which would provide a net gain in jamming protection when
combined with spread spectrum coding.

4) The possible use of a more protected frame synchronization
scheme in multichannel digital communications.

5) A means for exploiting the possibilities of coding gain
by the use of low rate error correcting codes as an alternative
to conventional spread spectrum.

6) Techniques for adaptive null-steering antennas to provide
significant AJ protection independently of any spread spectrum
gain.

7) Various frequency hopping techniques and novel ways to
remove some of the disadvantages of this method of achieving

spread spectrum for jamming protection.

The choice of topics was based on the special constraints imposed on systems design for tactical communications in general and on multichannel radio communications in particular. For example, the communication nodes can be expected to be small, stationary and sometimes located in isolated positions; available bandwidth for spreading is not expected to be large (if it is available at all); the jammers are not expected to be highly sophisticated even though they can be expected to know the weaknesses and vulnerabilities of the communications systems; and it is assumed that it is more desirable to reduce the grade of service for communications under a jamming threat in some proportion to that threat than it is to allow the system to degrade totally below acceptable levels.

The analysis carried out to date shows significant promise for achieving the goal of tactical ECCM within the constraints imposed. In particular, we can draw conclusions, with some confidence, about the effectiveness of the techniques studies in this final report. They are:

1) It is possible to effect a rational and reasonable re-allocation of communication channel capacity assignment which meets the threats, imposed by a jammer, of possibly disrupting all communications. A method for reallocation of channel capacity based on offered traffic and priority classes of users has been proposed and has been illustrated with examples. These examples show that a) pooled channel capacity requires fewer total channels

for a given traffic and grade of service; b) under jamming conditions it is possible to reassign the _availability_ of channels based on priority so that all users get protection and high priority users retain their grade of service (probability of being blocked) while lower priority users have this grade and service somewhat degraded. Furthermore, it is possible to monitor bit error rate by using the methods of extreme value theory to obtain rapid indication of the threat of being jammed so that appropriate reallocation of channel capacity can be effected. The combination of channel bit error rate monitoring and the method of reallocation based on priority users can result in a dynamic and adaptive scheme for making most effective use of the available bandwidth based directly on _real_ _time_ traffic needs, priorities and jamming conditions.

2) Detailed calculations have been made which now allow us to assess the tradeoffs of using bandwidth efficient (high bits per second per Hertz) signalling alphabets to achieve greater protection gain against the greater vulnerability to interference and jamming that these schemes invariably have. Detailed equations were derived for FSK, and M-ary QASK including the effects introduced by Gray-coding the constellation. For straight noise jamming it appears that a) reduced MARK-SPACE frequencies below the orthogonality condition in FSK is counter-productive; b) QPSK appears to be the best tradeoff when J/S is high (40dB). The probability of bit error equations can be used to investigate for the particular vulnerabilities present in multichannel systems where some bits in

a frame may be more important than others (e.g. frame synch).

Also the bit error rate rate results can be used to assess the value of dynamic allocation of capacity as described above, and it is demonstrated that gross benefits can accrue in performance by dynamic allocation when high jamming conditions are present. Detailed curves based on the computations are presented and can be used for other investigative and design purposes as well.

3) Low rate error correcting codes are a viable and often a very effective alternative to spread spectrum for jamming protection. We have demonstrated several particular classes of codes one of which is related to the linear PN sequences which are a) easy to encode; b) relatively easy to decode; c) for hard decision decoding exhibit good bit error rates near $E_b/N_o = 10$; d) perform better than DS spread spectrum with the same spreading factor; e) provide the added bonus of detecting virtually all errors. Numerical calculations of performance have been carried out to verify these conclusions. Asymptotic analytic bounds have been found which support the numerical results.

For soft decision decoding, these codes are effective in providing coding gain for low E_b/N_o. The coding gain can be 10 dB or higher but would require banks of matched filters. Another class of low rate convolutional codes have been found which exhibit excellent performance and which show promise of

relatively easy decodability. There are distinct advantages of using low rate codes either in place of direct sequence spread spectrum or as an adjunct to any of the spread spectrum schemes being used now.

4) It is now possible to analyze the complete steady-state and transient behavior of a linear mean square (LMS) algorithm adaptive null steering antenna array. It should therefore be possible to a) determine the dynamics of the array when it is in the process of being jammed; b) analyze the vulnerability of the antenna array to "blinking" or periodic jammers; c) give insight to algorithm design which is simple, cost effective and achieves expected performance; c) provide a starting point for the understanding of array behavior under noise and barrage jammers, CW tone and burst tone jammers.

Properly designed adaptive antenna arrays offer great promise for achieving a high degree of jamming protection without expending additional bandwidth (which may not be available in the tactical environment. Furthermore, in principle, antenna arrays can be "front-ended" to existing equipment without major modification in modulation format, channelization, etc.

5) Frequency hopping systems offer good promise as an ECCM technique. Many ideas associated with frequency hopping need to be developed to exploit the full potential of FH. Alternate means of correlation at the receiver can be used to change the character of the "sliding correlation function" and permit sidelobeless

outputs. The effects of partial band jamming on FH can be devastating if FH modulation is approached naively since it allows a jammer to optimize a strategy for maximum communications damage. These problems can, however, be overcome by diversity, coding and other techniques.

Recommendations for Further Work

A. Investigative

* Refine algorithms for adaptive allocation of capacity under jamming threats. This will also require the development of an effective nonparametric technique for sensing jamming conditions and estimating BER "on-line".

* Explore the properties of the complete (low rate) convolutional codes with a specific view of defining a relatively simple algorithm for soft decision decoding of these codes.

* Investigate the periodic response of adaptive array antenna algorithms in order to uncover means for avoiding the vulnerability to blinking jammers. Since the analysis available appears to be able to handle multiple jamming signal approaching from different angles, this should be given particular attention.

* Analyze further the behavior of adaptive arrays in noise and random jamming signals. This analysis has never been carried out until the work reported here. The payoff in understanding the response of the array to stochastic signals can be great since real jammers are best modeled by random signals.

B. Developmental

Based on the direct results of our research and also on the contact with other problems that we have had in the course of our investigations, we recommend a number of development projects which would hopefully exploit the theoretical ideas discovered and take advantage of the opportunities that a successful VHSI and VLSI endeavor will produce by the mid 1980's.

- Implementation of all the signal processing functions of direct sequence spread spectrum receiver on a chip. This would include correlators, matched filters for acquisition, logic, clocks, sequence generators, data detectors and alarms.

- Implementation of a digitally controlled frequency synthesizer in VLSI. This would be a challanging task since frequency synthesis functions are predominantly analog such as filters, mixers, etc. The new analog integrated circuit concepts such as capacitor switching may be invoked here. Success in this venture would make frequency hopping systems much more palatable.

- Develop an integrated circuit matched filter bank for the Reed-Muller codes (PN Sequence Codes) and other block codes shown to be effective over Gaussian Channels. This would require the equivalent of about 1000 tapped delay line systems and could conceivably be done in VLSI. The achievement of integrated matched filter banks could result in substantial coding gain as shown in this report. Additionally, such a development could be

used for rapid acquisition and other signal processing objectives in ECCM.

• Develop the front end and signal processing functions of an adaptive array. At present adaptive arrays are expensive since they require multiply many front ends and the signal processing that goes with each. One far out possibility is to produce a VHSI "sampled-data" adaptive array.

• Develop a convolutional decoder in a very small package. This might be a maximal likelihood (Viterbi) decoder of moderate constraint length (3-4) on a chip or a hard decision decoder based on a possible algebraic algorithm for the complete (low rate) codes described in this report.

• Develop "Smart" multiplexors for digital multichannel communications which can switch traffic, assign priorities, operate at variable data rates and variable channel allocations. This would require the multiplexor to have a microcomputer; direct access memory and programming capability. One of many applications of such a smart multiplexor would be to implement a priority based adaptive channel allocation scheme as described in this report.

Appendix A. Spread Spectrum Background - Frequency Hopping Systems

A.1. Introduction

The ECCM technology has produced, over the past 20 years a plethora of ideas for jam-resistant communications. Some indication of the diversity of the proposals is provided by examination of the list below:

1. Direct Sequence Pseudo-Noise (PN)

2. Frequency Hopping (FH)

3. Frequency-Time Dodging

4. Time Dodging Burst

5. Chirp Techniques

6. Interleaved Error Correcting Coding

7. Moving Notch Filters

8. Adaptive Signal Cancellers

9. Receiver Blanking Techniques

10. Adaptive Antenna Nulling
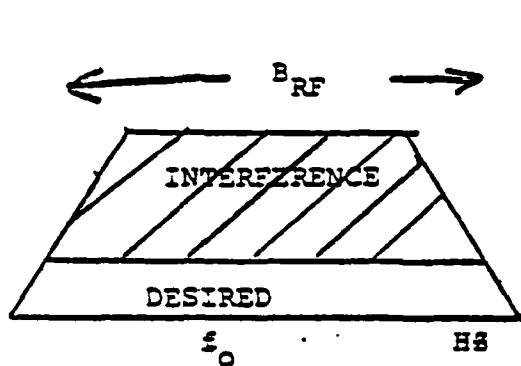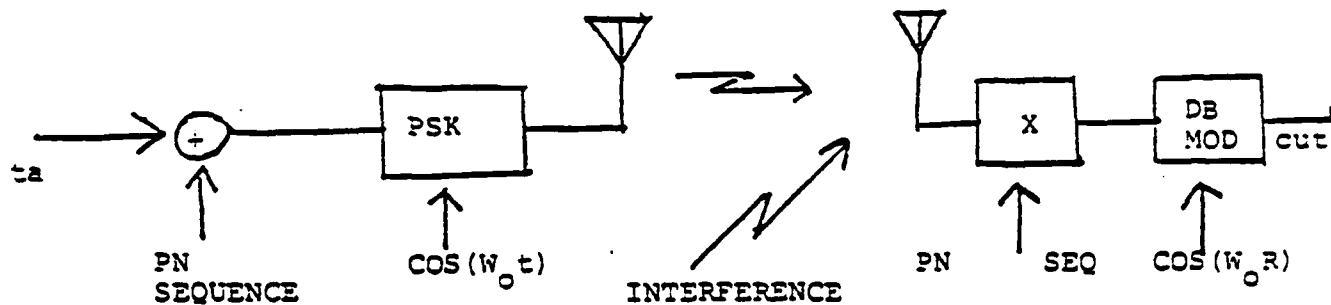
11. Weak Signal Amplifiers

The first six items on this list involve some or a large degree of spectral spreading so that the amount of bandwidth required per bit per second of raw data is larger than would normally be necessary. The other techniques, while not utilizing the bandwidth as a resource for achieving jamming protection require a priori knowledge of the nature of the jammer or, in the case of adaptive null steering antennas, an array receiver. In this appendix, a review of the analysis and major design issues for frequency hopping will be presented.

- 171 -

On the premise that, in any event, certain resources must be expended to achieve anti-jamming capability, it is important to first address the question of how one measures ECCM effectiveness. Perhaps the most commonly used measure of protection against jamming is a quantity known as the processing gain. In the following pages, after defining Processing Gain and its concomitant system measure, Jamming Margin, we will show how singularly misleading this measure can be in general and, in particular, when assessing the performance of frequency hopping systems.
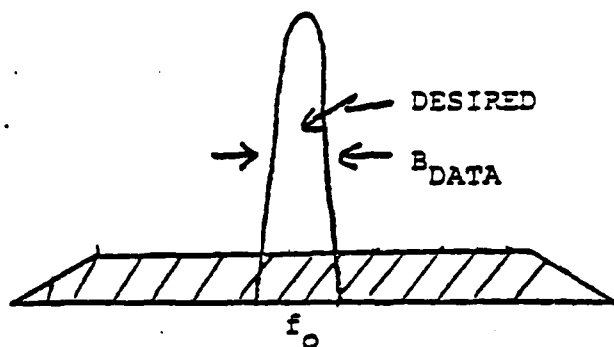
A.2. Processing Gain

The notion of processing gain (P.G.) derives from the fact that a signal to noise (jammer) ratio improvement can be obtained by a receiver which possesses a replica of the wideband modulation carrier by correlation or matched filter processing. This is shown in Figure A.1a where, for illustration purposes, the broad band process in a pseudo-noise (PN) sequence. The pseudo-noise at the transmitter causes the transmission to be spread and a bandwidth B_{RF} which is large compared to the data bandwidth, B_D. Correlating the received signal with a replica (time synchronized) of the transmitted spreading signal causes the resulting signal to "coherently collapse" in bandwidth to that of the data spectrum.

Subsequent post-correlation filtering then permits all of the data spectrum to be recovered while accepting only a

PSK

DB
MOD | cut
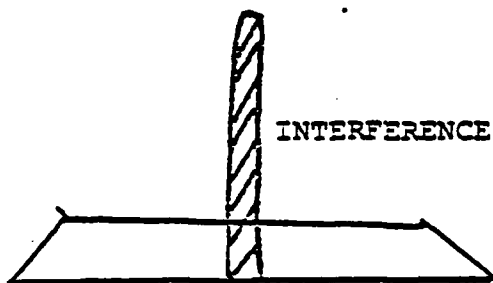
ta

PN
SEQUENCE

$COS(W_o t)$

INTERFERENCE

X

PN   SEQ $COS(W_o R)$

B_{RF}

INTERFERENCE

DESIRED

f_o HZ

PRE-CORRELATION

DESIRED

B_{DATA}

f_o

POST-CORRELATION

WIDEBAND   INTERFERENCE

$$P.\ G.\ =\ \frac{B_{RF}}{B_{DATA}}$$

INTERFERENCE

B_D

B_{RF}

NARROWBAND   INTERFERENCE

$$P.\ G.\ =\ \frac{B_{RF}}{S_{DATA}}$$
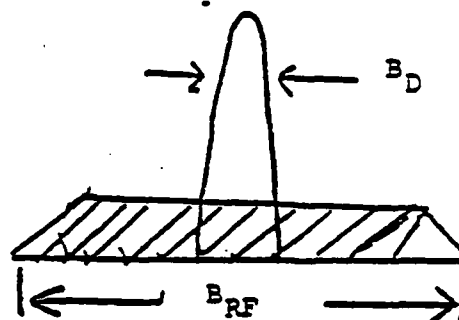
ex t_C = 1μs B_{AF} = 2.0 MHZ

          T = 1ms, B_D = 2 KHZ   40dB

Figure A.1a SPREAD SPECTRUM PROCESSING GAIN

- 173 -

fraction B_{RF}/B_D of the interfering signal. This description
is one of many ways of viewing the achievement of the pro-
cessing gain and is easily formalized at baseband as follows:

$$\text{Transmitted signal} = s(t) = X(t) \ d(t)$$

$$\text{Where} \qquad X(t) = \text{pseudo noise}$$

$$d(t) = \text{data}$$

Receiver correlation output $R(t)$:

$$r(t,\tau) = [X(t)d(t) \cdot X(t-\tau)] * h(t)$$

Where $* h(t)$ denotes convolution with the impulse response
of the receiver filter and τ is the local time offset which
is adjusted to zero during the process of acquisition and main-
tained near zero by tracking loops. The receiver output is
then

$$r(t) = d(t) * h(t)$$

or, in terms of the Fourier transform,

$$R(f) = D(f) \cdot H(f)$$

so that if $D(f) = 0$ for $|f| > B_D$ it is possible to choose

$$H(f) = \begin{cases} 1 & ; \ |f| < B_D \\ 0 & ; \ |f| > B_D \end{cases}$$

and thus

$$r(t) = d(t) \quad \text{[data recovered]}$$

An interfering signal, not matched to $x(t)$ however will yield:

$$r_1(t) = N(t) \cdot X(t) * h(t)$$

Where, if $N(t)$, the interfering signal is a narrowband spot
frequency jammer, say, then the mean squared power out of the
correlation receiver becomes

- 174 -

$$\bar{r}_1{}^2 = \int_{-\infty}^{+\infty} S_x(f) |H(f)|^2 df$$

and if $S_x(f)$, the power spectrum transmitted bandwidth B_{RF}; we have

$$S_x(f) = \frac{\sigma_N^2}{2B_{RF}}$$

and finally

$$\bar{r}_1^2 = \sigma_N^2 \cdot \frac{B_D}{B_{RF}}$$

Where σ_N^2 = input interference power which we observe to have been reduced by the processing gain, i.e.

$$P.G. = \frac{B_{RF}}{B_D}$$

While the processing gain concept and the analysis leading to it is valid, the number has only a second moment interpretation. That is, the processing gain can only tell us how much advantage we gain in <u>average</u> interference <u>power</u> that finally is allowed to pass through the correlation receiver. By itself, the processing gain cannot be used as a measure of performance in digital communications. However, with some care, and under certain circumstances it is possible to obtain meaningful anti-jamming performance measures from the processing gain. For example, for long binary PN sequences which, when filtered behave like Gaussian noise, the probability of error can be estimated by:

$$P(\varepsilon) = f(\gamma \cdot P.G) \qquad (A.1)$$

Where

$$\gamma = \frac{S}{J} = \frac{\text{signal power}}{\text{jamming power}}$$

- 175 -

and $f(\cdot)$ depends on the modem e.g. for non-coherent FSK

$$f(x) = \frac{1}{2} \exp-x \qquad\qquad (A.2)$$

for coherent binary PSK

$$f(x) = \frac{1}{2} \operatorname{erfc} \sqrt{x} \qquad\qquad (A.3)$$

for binary DPSK

$$f(x) = \frac{1}{2} \exp-\frac{x}{2} \qquad\qquad (A.4)$$

as shown in Figure A.1b.

Thus, to achieve a bit error rate (B.E.R.) on $P(\varepsilon)$ of 10^{-5} we require

$$x = \gamma \cdot PG = 9.6 \text{ dB}$$

now if $PG = 30$ dB (1000:1 bandspread) then we could tolerate J/S of $9.6 - 30 = 20.4$ dB. In other words, the jammer would require more than 30 dB or 100 times more power at the receiver before the performance of B.E.R. $= 10^{-5}$ would be degraded! Indeed, except for the ommission of implementation losses and the approximation of the filtered PN sequence as Gaussian noise, this calculation can be used to compute a direct performance measure, i.e. B.E.R.

The protection ratio afforded by any spread spectrum system is theoretically determined by B_{RF}/R_D where B_{RF} is the radio frequency bandwidth and R_D is the data rate (bps) (and is also the nominal bandwidth B_D that would be required to transmit binary data without spreading). The protection ratio is simply the theoretical advantage on processing gain that a desired signal (that is matched in frequency, code and time) has over a jamming signal (wideband noise over the band, or a spot,

Figure A.1b BINARY BER
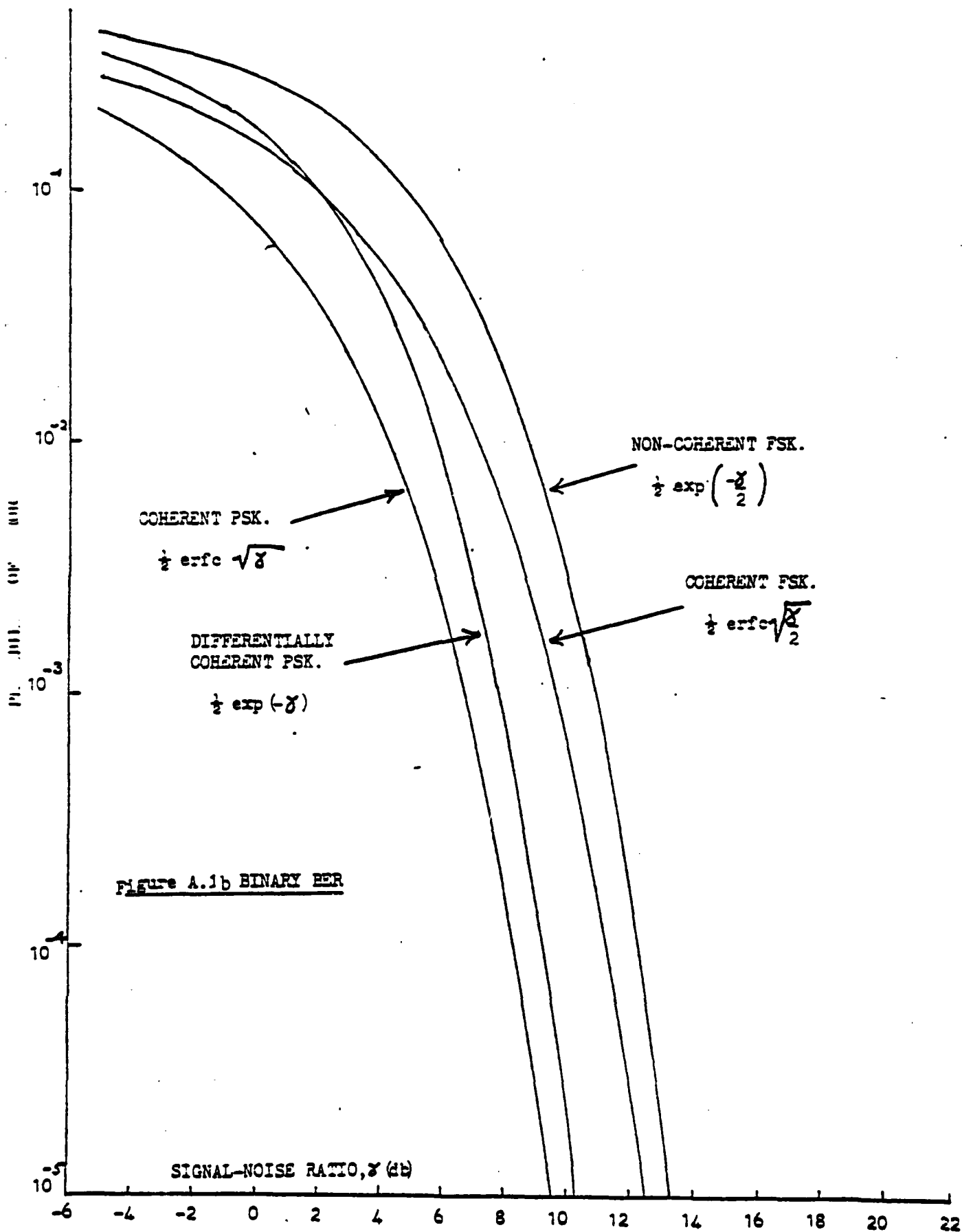
single or multiple frequency) that is not properly informed.
That is, for the same received signal power the desired signal
will emerge, after processing, with B_{RF}/R_D more power than the
jamming signal. Or, to put it another way, for the jammer to
affect a response equal to that of the desired signal, he would
have to provide B_{RF}/R_D more power than the desired signal. Thus
if $G_r = \frac{B_{RF}}{R_o}$ then (J/S) out $\sim G_p^{-1}(J/S)$ in, where (J/S) out is the
output jamming to signal power.

The theoretical processing gain, however, is not, by itself,
a measure of how well the system is capable of performing in a
jamming environment. For this purpose we usually introduce
the ideal of _jamming margin_ (M_j). This is the residual advantage
that the system has after we subtract (in dB):

1. The minimum $(S/N)_{out}$ necessary to achieve the transmission
performance required. (Typically, this would be specified in
terms of E_b/N_o, the energy (joules) per information bit divided
by the "noise" spectral density in watts/Hz required after the
processing gain to achieve a specified probability of error).

2. The implementation loss (L_I) due to imperfect timing,
correlation, etc.

3. The loss in protection (L_J) that would be incurred
by a more intelligent jammer that takes advantage of knowledge
of the system which allows him a greater degree of penetration
than that obtainable by brute force CW or noise jamming.

Thus in decibels,

$$M_j = G_p - (\frac{S}{N})_{out}^* - L_I - L_J$$

For example, if $B_{RF} = 500$ KHZ, R - 16 Kbps,

then

$$G_p = \frac{50 \times 10^3}{16 \times 10^3} = 9.72 \text{ dB}$$

Typically, $L_I = 1.5 - 2$ dB

$$\frac{E_b}{N_0} = (\frac{S}{N})_{out} = 10 \text{ dB (For BER } = 10^{-5}, \text{ BPSK no coding)}$$

say, $L_J = 2$ dB due to partial band jamming

Then

$$M_j = 9.72 - 2 - 10 - 2 = -4.28 \text{ dB}$$

This means that the system will operate within the prescribed
performances (BER $\leq 10^{-5}$) up to the point when the jammer
power is as much as 4.28 <u>lower</u> than that of the desired signal
at the receiver input. If the jammer power is larger than
this, than although the processing gain still appears to favor
the desired signal, the performance will drop below the require-
ments (BER $\leq 10^{-6}$).

Several additional points should be made:

a. In a complicated signal structure involving FH, PN,
TH, G_p may not be B_{RF}/R_D but it certainly will not be greater
(against a noise jammer). G_p, in fact, implies a reference
jammer which is usually taken as stationary Gaussian noise
all of whose power is uniformly spread across the bandwidth in
question. This defines N_0 in watts/Hz once B_{RF} is specified.

b. The loss due to an intelligent jammer depends strongly
on the perceived threat. That is, it depends on how much value
an enemy is willing to place on taking advantage of his knowledge
of the system (it is always a tacit understanding that an enemy

knows the system design and the only thing denied to him is the code of the day). If the enemy choses not to take advantage of this knowledge then his use of brute force noise or CW will cost him L_J dB (or, we can pick up the L_J dB to our jamming margin or reduce our B_{RF} requirements, etc.)

c. The $(\frac{S}{N})_{out}$ can be reduced (thereby increasing M_J for specified G_p or vice versa) by the use of coding. Typically, we can pick up anywhere from 2-5 dB with coding. That is, to achieve a specified probability of error it may require 2 to 5 dB less $(\frac{S}{N})_{out}$ and this may be used to increase M_J or reduce G_p required for a given M_J. It should be noted that for a frequency hopping system it is usually essential that some coding be used because the protection ratio provided by that method, being effectively an evasion tactic, may not be effective even against a spot jammer since all the jammer must do is to degrade the system performance beyond the acceptable range. Thus if a frequency hopping system uses 100 frequencies, a spot jammer sitting on one of them will cause an average error rate of 10^{-2}. Even though it appears that the protection gain achievable is 20 dB, the system is totally vulnerable. Coding forces the jammer to occupy many frequency slots and since he doesn't know the pattern, wiping out just a few of the code symbols still leaves the full power of the code to recover the data. Since most block codes (including Reed-Solomon M-ary symbol codes) are effective provided not too many symbols are caused to be in error in a given block, it is usually also necessary to interleave the data before transmission and deinterleave it at the receiver so that bursts of errors

- 130 -

caused by a high power pulse jammer is distributed among many blocks and the code retains the effectiveness.

d. The jamming margin provided above is based on spread spectrum signal structure alone. Additional and significant (10 - 20 dB or more) anti-jamming protection may be obtained from adaptive null steering antennas which discriminates against a jammer because of the physical separation of the desired transmitter and the jammer. This technique does not, in principle, require bandspreading but it is most naturally implemented in conjunction with PN spread spectrum which provides a necessary reference signal.

e. Finally, the actual jamming margin, M_J required is an operational specification that is, it is determined by the perceived jamming threat scenario. Thus, if a requirement of say, 12dB jamming margin is based on the assumptions of say, 2KW jammers deployed in a certain way and with 100 Watt friendly transmitters, then all other assumptions remaining the same, 3 dB of jamming margin can be gained by increasing the power to 200 watts per transmitter. The total electro-magnetic "clutter" caused by this move would, of course, increase by 3 dB but if this could be tolerated by systems occupying the same band then one could presummably obtain the same performance against the specified jamming threat with 3dB less processing gain (or one half the required B_{RF} if we choose to trade on that).

In this appendix we hope to introduce some of the properties and problems associated with frequency hopping and especially

as they relate to the possible use of such systems to minimize jamming vulnerability or to maximize jamming margin as defined above. The objective is to record some analysis which is not generally available or known and to point to the possible pitfalls that await naive application of formulas.

A. 3. Frequency Hopping

Frequency hopping is an evasion technique by which the "instantaneous" frequency of a carrier is changed pseudo-randomly every T sec. 1/T is the hopping rate and the pattern is illustrated in Figure A.1c where the frequencies are chosen from the set $\{f_i\}$ corresponding to pseudo-random code i. Because the spectral occupancy of a finite burst of carrier is approximately 1/T, it is necessary to space the separation between frequencies by the order of 1/T. Thus if there are N frequencies in the set over which hopping takes place, then B_{RF}, the total range is N/T as shown in Figure A.2 where the time-frequency occupancy is shown.

A typical implementation of the Frequency Hopping (FH) transmitter is shown in figure A.3. A digitally controlled Frequency Synthesizer (to be discussed later) is driven by a PN generator to generate a pseudo-random hopping pattern. A common way to impart data is to allow the least significant bit (LSB) of the frequency synthesizer control input to cause frequency shift keying of the hopped signal in order to generate mark and space symbols. The FH transmitter could either be coherent and preserve relative phase in transitions or, as is
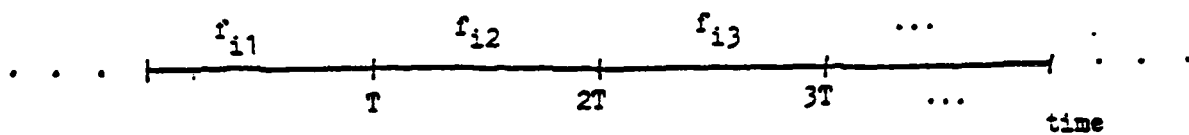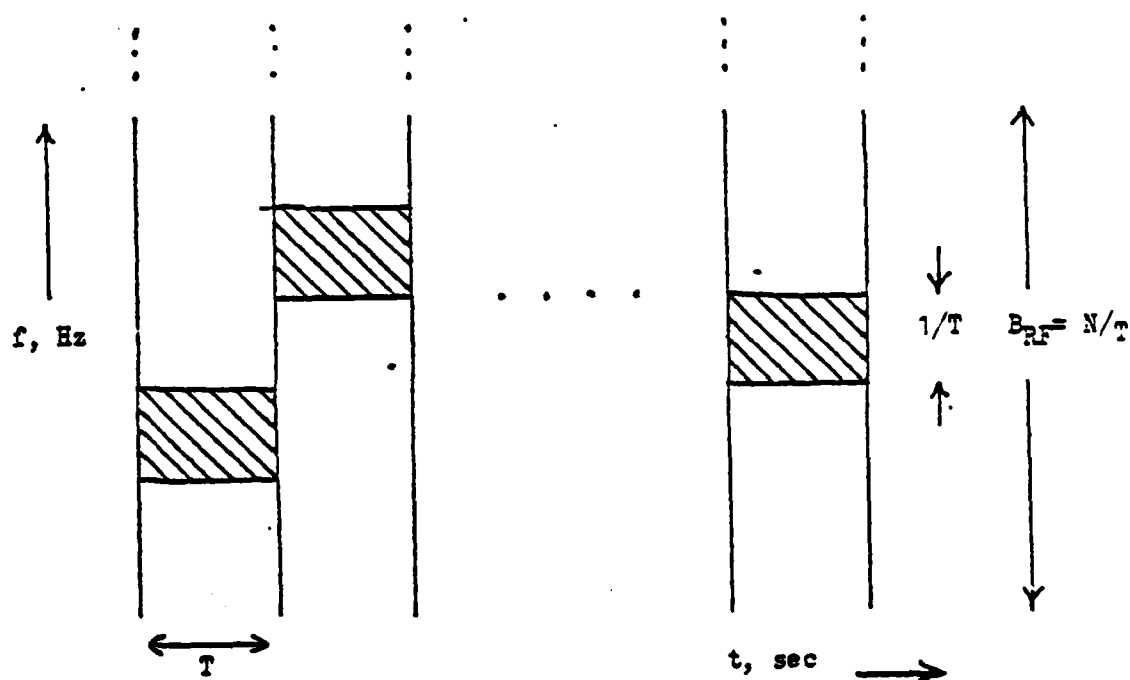
Figure A.1c - Hopping Pattern

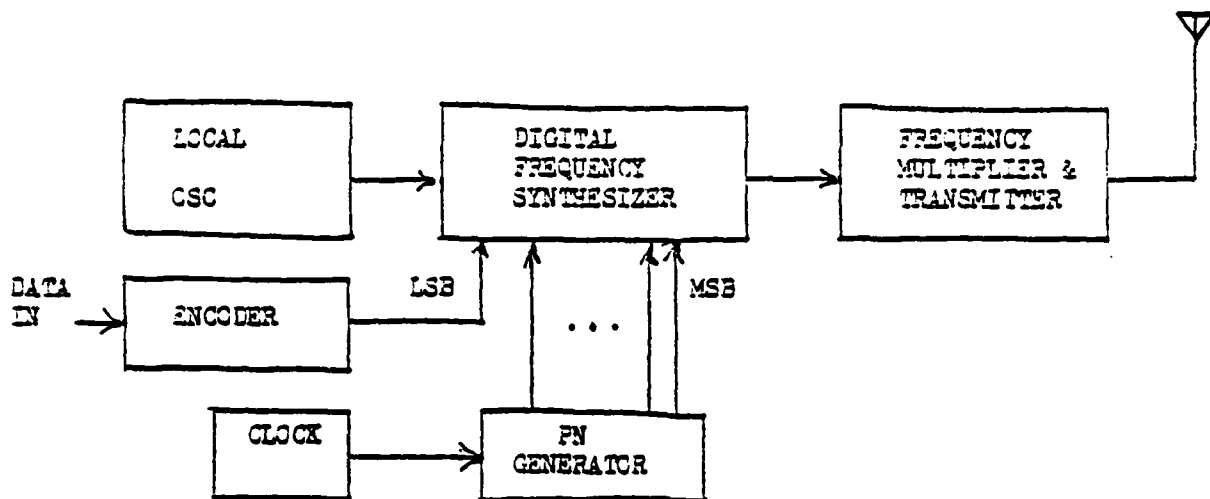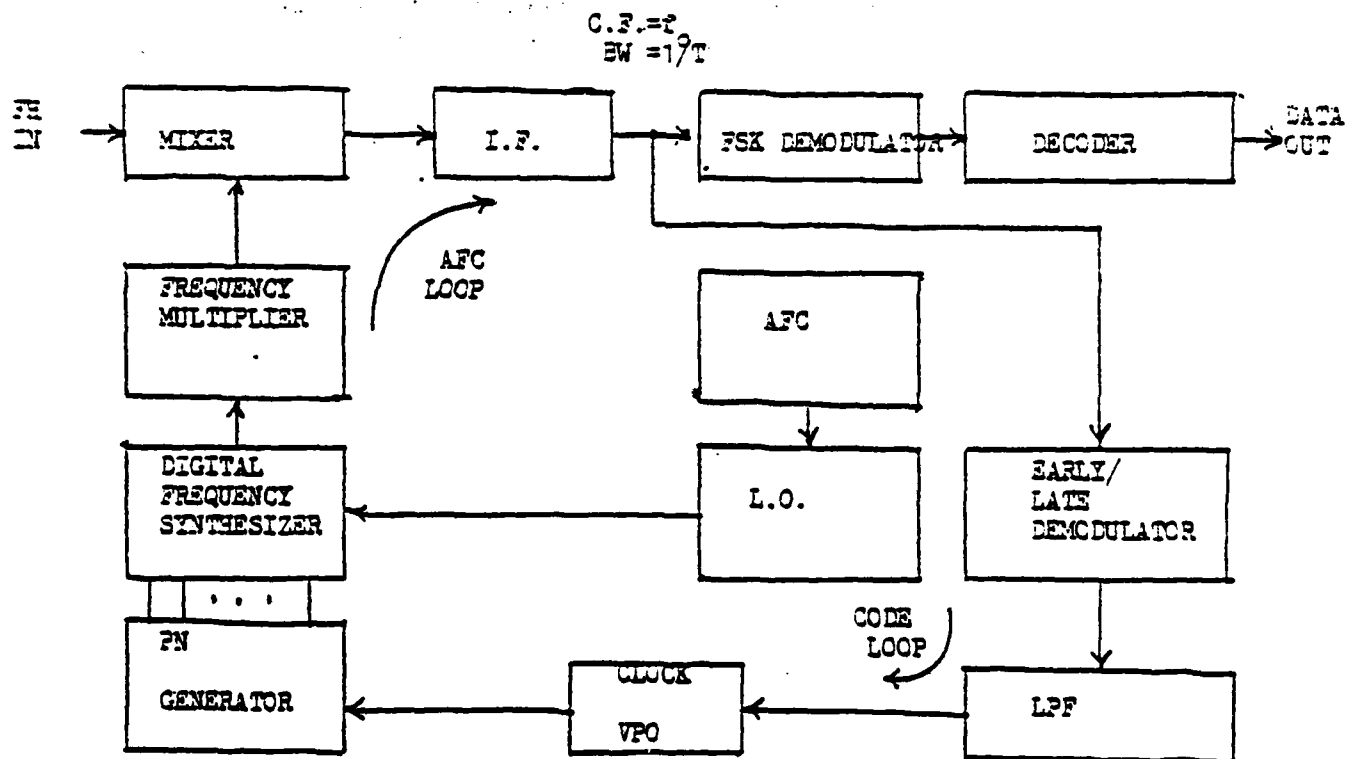Figure A.2 - Time-Frequency Occupancy

Figure A.3 - FH Transmitter

$$C.F. = ?$$
$$BW = 1/T$$

Figure A.4 -, FH Receiver

- 184 -

more commonly the practice, the transmitter phase is not pre-
served and we have non-coherent FH. The FH receiver is more
complicated since provision must be made for acquisition and
tracking. A simplified block diagram is shown in figure A.4.
The process of demodulation after code acquisition has taken
place is illustrated by examining Figure A.5 where the local
synthesizer effectively "dehops" the signal to a standard
FSK signal for demodulating the data.

In order to fully appreciate the properties of FH for
purposes of acquisition and spectral analysis, we shall
derive the auto correlation function and the power spectral
density of a coherently hopped signal, the hopping pattern of which
is assured to be random.

A.4. Autocorrelation Function of Coherent Frequency Hopping

Consider a wave form generated by selecting at random one
of 2N sine functions $\sin (n\Delta wt)$ $n = \pm K, \pm(K + 1)...\pm(K + N - 1)$
every T seconds. Assume further that $K >> 1/Tw$ so that there are
many r.f. cycles in any one interval, T.

The total waveform can be represented by

$$s(t) = \sum_{n=K}^{K+N-1} P_T^{(n)} (t) \sin n\Delta wt \qquad (A.5)$$

where $P_T^{(n)}$ is a random pulse train of 1, o, and -1 so that

$$av. < P_T^{(n)} > = 0 \qquad (A.6)$$

and

$$av. \overline{P_T^{(n)}}^2 >= 0 \qquad (A.7)$$

- 185 -

DATA: 1   1   0   1   0   1

XMTD: $f_{11} + f$ $f_{12} + f$ f_{13} $f_{14} + f$ f_{15} $f_{16} + f$

$\xleftarrow{\hspace{1cm}}\xrightarrow{\hspace{1cm}}$

T

RCVR
SYNTH: $f_{11} + f_0$ $f_{12} + f_0$ $f_{13} + f_0$ $f_{14} + f_0$ $f_{15} + f_0$ $f_{16} + f_0$

I.F.
out: $f_0 + f$ $f_0 + f$ f_0 $f_0 + f$ f_0 $f_0 + f$

FSK

DEMOD: 1   1   0 1   1   0   1
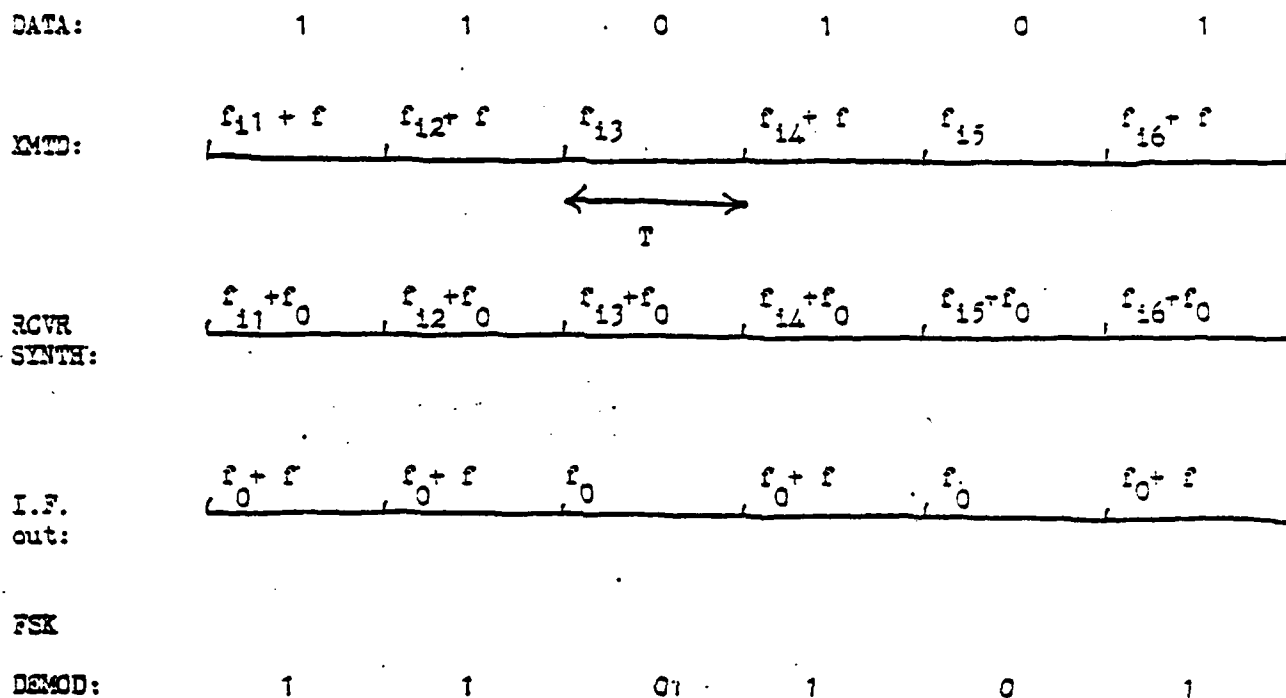
Figure A.5. - FH Dehopping

and $\Delta f = 2\pi\Delta w$ is the frequency spacing.

A typical $s(t)$ is shown in Figure A.6. This signal may be generated by the scheme shown in Figure A.7.

The auto correlation function $\phi_s(\tau)$ of $s(t)$ taking into account the fact that $K \gg 1$ is:

$$\phi_s(\tau) = \frac{1}{2} \sum_{n=K}^{K+N-1} \phi_{P_T}(n) \cos n\Delta w\tau \qquad (A.8)$$

Now $\phi_{P_T(\tau)}^{(n)}$ is the same for all n and is readily seen to be

$$\phi_{P_T}(\tau) = \frac{1}{N} \left(1 - \frac{|\tau|}{T}\right) \; ; \; |\tau| \leq T \qquad (A.9)$$

$$= 0 \qquad \qquad ; \; |\tau| > T$$

Hence,

$$\phi_s(\tau) = \frac{1}{2N} \left(1 - \frac{|\tau|}{T}\right) \sum_{n=K}^{K+N-1} \cos n\Delta w\tau \; ; \; |\tau| < T \qquad (A.10)$$

or

$$\phi_s(\tau) = \frac{1}{2N} \left(1 - \frac{|\tau|}{T}\right) \left[\sum_{n=0}^{K+N-1} \cos n\Delta w\tau - \sum_{n=0}^{K-1} \cos n\Delta w\tau \right]$$

The finite trigonometric sum has a closed form expression* so that

$$\phi_s(\tau) = \frac{1}{2N} \left(1 - \frac{|\tau|}{T}\right) \cdot \frac{\cos \frac{K+N-1}{2}\Delta w\tau \sin \frac{K+N}{2}\Delta w\tau}{\sin \frac{\Delta w\tau}{2}}$$

$$- \frac{\cos \frac{K-1}{2}\Delta w\tau \sin \frac{K}{2}\Delta w\tau}{\sin \frac{\Delta w\tau}{2}}$$

_____

*See R.P. Adams Smithson, Mathematical Formulae, p. 81.
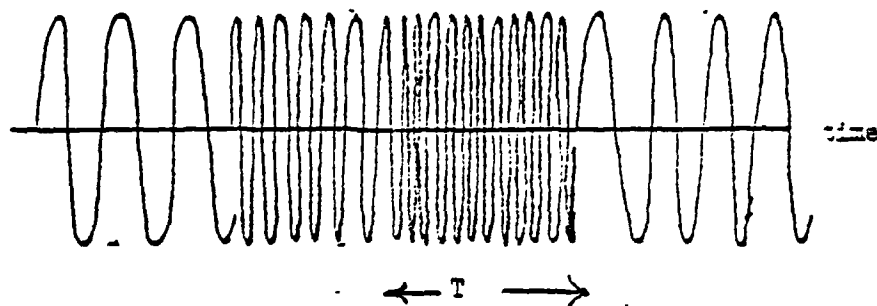
- 187 -

time

$\longleftarrow$ T $\longrightarrow$

Figure A.6 - Frequency Hopped Signal
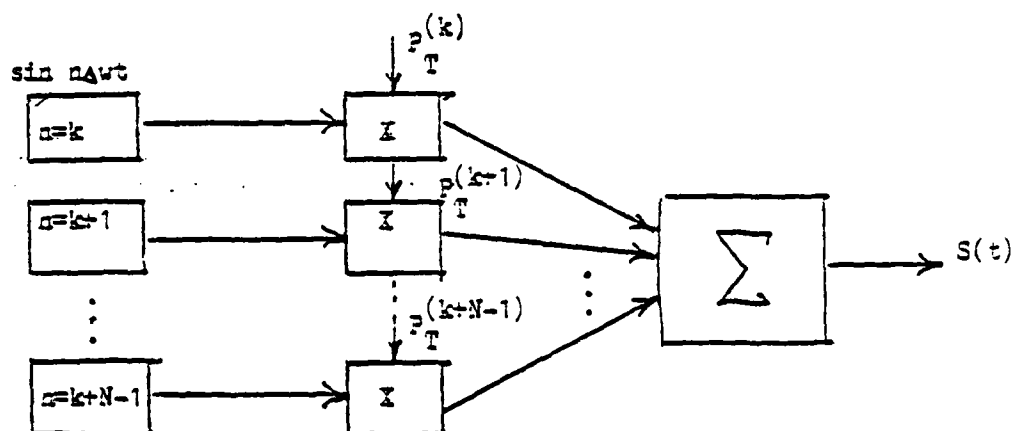
Figure A.7 - Frequency Hopping Generator
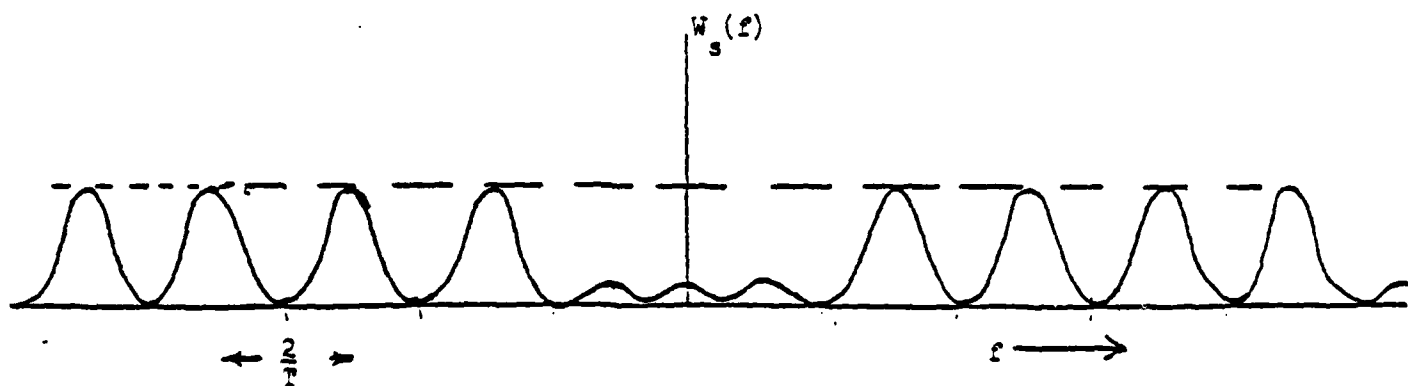
Figure A.8 - Power Spectral Density; N=4, $\Delta f = \frac{2}{T}$

$$= \frac{1}{4N} \left(1 - \frac{|\tau|}{T}\right) \frac{[\sin \frac{2K+2N-1}{2} \Delta w\tau - \sin \frac{2K-1}{2} \Delta w\tau]}{\sin \frac{\Delta w\tau}{2}}$$
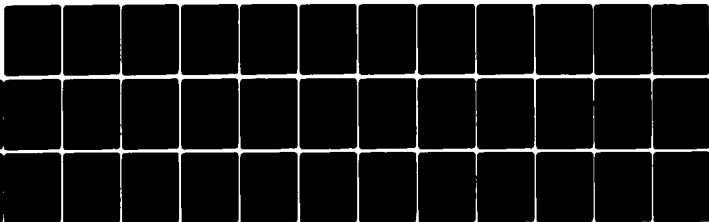
$$\phi_s(\tau) = \frac{1}{2N} \left\{ \frac{(1 - \frac{|\tau|}{T})}{\sin \frac{\Delta w\tau}{2}} \sin \frac{N}{2} \Delta w\tau \right\} \cos \frac{2K+N-1}{2} \Delta w\tau; |\tau| \le T \qquad (A.11)$$

$$= 0 \qquad\qquad\qquad\qquad ; |\tau| \le T$$

Since $(2K+N-1) \gg N$ the bracketed term is recognized as the envelope of the autocorrelation function. Furthermore, when $N \gg 1$ then

$$\frac{1}{2N} \frac{(1 - \frac{|\tau|}{t})}{\sin \frac{\Delta w\tau}{2}}$$

is in the envelope of the envelope.

Equation (A.11) can be normalized by dividing by $\phi_s(0) = 1/2$, and setting $\tau/T = 0$. Then we get

$$\frac{\phi_s(\theta)}{\phi_s(0)} = \frac{1}{N} \left(1 - |\theta|\right) \frac{\sin \frac{N}{2} \Delta wT\theta}{\sin \frac{1}{2} \Delta wT\theta} \cos \frac{2K+N-1}{2} \Delta wT\theta \qquad (A.12)$$

$$= 0 \qquad\qquad\qquad\qquad ; \quad |\theta| \ge 1$$
$$\qquad\qquad\qquad\qquad\qquad |\theta| > 1$$

Some interesting results are easily obtained form the bracketed function. For example the first significant sidelobe occurs when $\frac{N}{2} \Delta wT\theta = \frac{3\pi}{N\Delta wT}$. Then if N is large,

$$\left\{ \frac{\phi_s(\tau)}{\phi_s(0)} \right\} = \frac{1}{N} \left(1 - \frac{3\pi}{N\Delta wT}\right) \cdot \frac{1}{\frac{3\Delta\pi}{2N}} \qquad (A.13)$$

$$\to \frac{2}{3\pi} = 0.212 \text{ (N very large)}$$

where $\left(\frac{\phi_s(\tau)}{\phi_s(o)}\right)$ is the bracketed envelope factor of the normalized autocorrelation function (A.13).

In order to avoid a large spurious peak, it is necessary that

$$\sin \tfrac{1}{2} \Delta w T \tau \neq 0 \qquad 0 \leq \theta < 1$$

It is sufficient that

$$\tfrac{1}{2} \Delta w T < \tfrac{\pi}{2}$$

or,

$$\Delta w < \tfrac{\pi}{T} \text{ or } \Delta f < \tfrac{1}{2T} \tag{A.14}$$

There are 2N lobes and an equal number of nulls.

A.5. Spectral Density

The power spectral density is obtained directly from the Fourier transform of (6) The Fourier transform of a typical term is

$$F\{\tfrac{1}{N}(1 - \tfrac{|\tau|}{T}) \cos nw_n T\} = \tfrac{1}{4}(W_{p_T})f - n\Delta f) + W_{p_T}(f + n\Delta f))$$

where W_{p_T} is the Fourier transform of $\phi_{p_T}(\tau)$

Now,

$$F\{\phi_{p_T}(\tau)\} = F\{\tfrac{1}{N}(1 - \tfrac{|\tau|}{T}) \qquad |\tau| < T$$

$$= \tfrac{T}{4N}(\tfrac{\sin \pi f T}{\pi f T})^2 \qquad |f| < \infty$$

Hence the two-sided spectral density of the frequency-hopped signal is:

$$W_s(f) = \frac{T}{16N} \sum_{n=K}^{K+N-1} (\frac{\sin \pi T(f-n\Delta f)}{\pi(f-n\Delta f)T})^2 + (\frac{\sin \pi T(f+n\Delta f)}{\pi(f+n\Delta f)T})^2 \qquad (A.15)$$

A typical spectrum for the case $\Delta f = \frac{2}{T}$. $N=4$ is shown in Figure A.8.

Notice that

$$\int_{-\infty}^{+\infty} W_s(f)df = \frac{1}{2} = \phi_s(o)$$

which is the mean-squared power of the signal.

Example

For the case $N = 16$ and the frequency separation Δf is $\frac{2}{T}$, the envelope normalized autocorrelation function $\{\phi_{16}(\theta)\}/\dot{\phi}_{16}(o)$ is plotted in Figure A.9. The dotted curve is the envelope of the sidelobe peaks. The first significant sidelobe has a normalized amplitude of 0.212 which was shown to be the maximum possible. The reason for the large central peak is due to the fact that Δf does not satisfy condition (A.14). Physically, the peak occurs because the frequencies chosen all have a period $\frac{1}{n\Delta f}$ which is an exact *even* submultiple of the quantizing period T. Therfore at exactly a shift of $\tau = T/2$ or $\theta = 1/2$, all the frequencies are in phase again. This central peak would not occur if $\Delta f = 1/T$.

The curve of Figure A.9 can be applied directly to a practical frequency hopping system. For example, it applies to a system using 16 frequencies separated by 100 KC and hopping at a 50 KC rate.

The large centralpeak of Figure A.9 would be extremely undesirable

Figure A.9 - ENVELOPE OF THE AUTOCORRELATION FUNCTION OF A

QUANTIZED FREQUENCY TRANSMISSION

$N=16$

$\Delta f_i = \frac{2}{T}$

$$\cdots \frac{1}{b} (1-\theta) \frac{1}{\sin 2\pi \theta}$$

$$\frac{1}{16} (1-\theta) \frac{\sin 2\pi (16)\theta}{\sin 2\pi \theta}$$

N Frequencies Spaced at

$$\theta = \frac{1}{T}$$
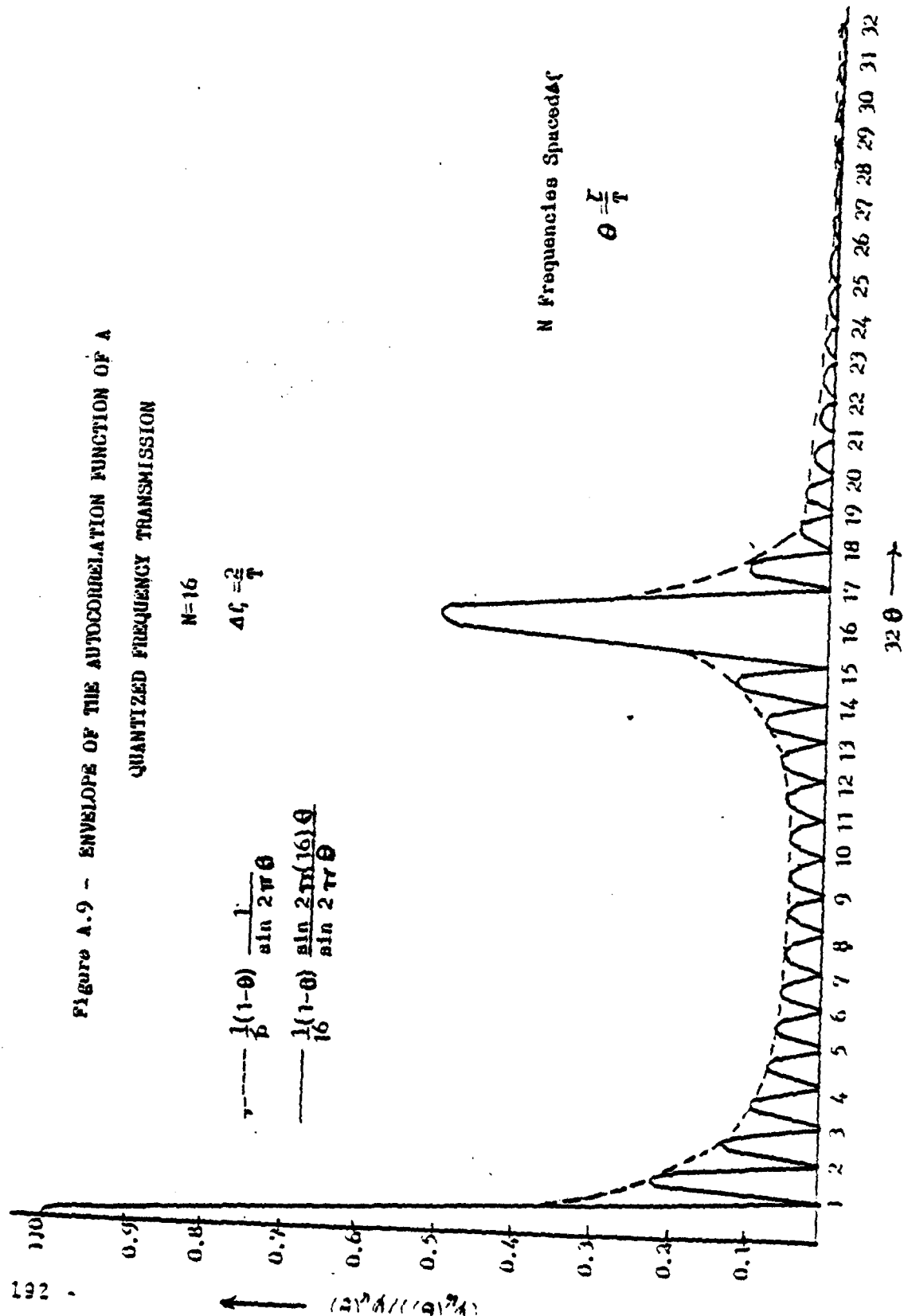
for acquisition since a false lock condition would occur with high probability. This coupled with the fact that the spectrum is not smooth (see figure A.8) neccesitates that we choose $\Delta f = \frac{1}{T}$ which insures 1) no spurious peak 2) smooth spectrum 3) orthogonality of the frequency bursts for good discrimination. For $\Delta f = \frac{1}{T}$, the autocorrelation function is shown in Figure A.10.

A.6. Alternate Derivation of the Autocorrelation Function

The derivivation given above is appealing because of the fact that it follows the physical means by which the signal is created. There are, however certain rough spots so that the alternate more rigorous derivation is warranted.

The autocorrelation of stochastic process $s(t)$ is defined as

$$\phi_s(t_1, t_2) = E[s(t_1) \times s(t_2)] = \iint p(s_1, s_2) s_1 s_2 \, ds_1 \, ds_2$$

where E is the statistical expectation and $p(s_1 \, s_2)$ is the joint probability density of $s(t_1)$ and $s(t_2)$ and

$$\phi_s(t_1, t_2) = E[s(t_1)] \times E[s(t_2)] = 0 \qquad |t_1 - t_2| > T$$

since

$$E[s(t_1)] = E[s(t_2)] = 0$$

If t_1 and t_2 are in the same interval, then $p(s_1, s_2) = 0$ except when $s(t_1)$ and $s(t_2)$ are on the same curve in which case $p(s_1, s_2)$ is the probability of the curve. If there are 2N curves

$$\hat{\phi}_s(t_1, t_2) = \sum_{n=1}^{2N} P_n s_n(t_1) s_n(t_2) \qquad \text{Where } P_n \text{ is the probability of the curve.}$$

Figure A.10 - ENVELOPE OF THE AUTOCORRELATION FUNCTION OF A
QUANTIZED FREQUENCY TRANSMISSION

$N=16$

$\Delta f = \frac{1}{T}$

$$-\; - \quad \frac{1}{16}(1-\theta)\frac{1}{\sin \pi \theta}$$

$$----- \quad \frac{1}{16}(1-\theta)\frac{\sin 16\, \pi \theta}{\sin \pi \theta}$$

N Frequencies Spaced Δf

$\theta = \frac{t}{T}$

$16\,\theta$

$\{\phi_\alpha(\theta)\}/\phi_\alpha(0)$

1.0 0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2 0.1

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
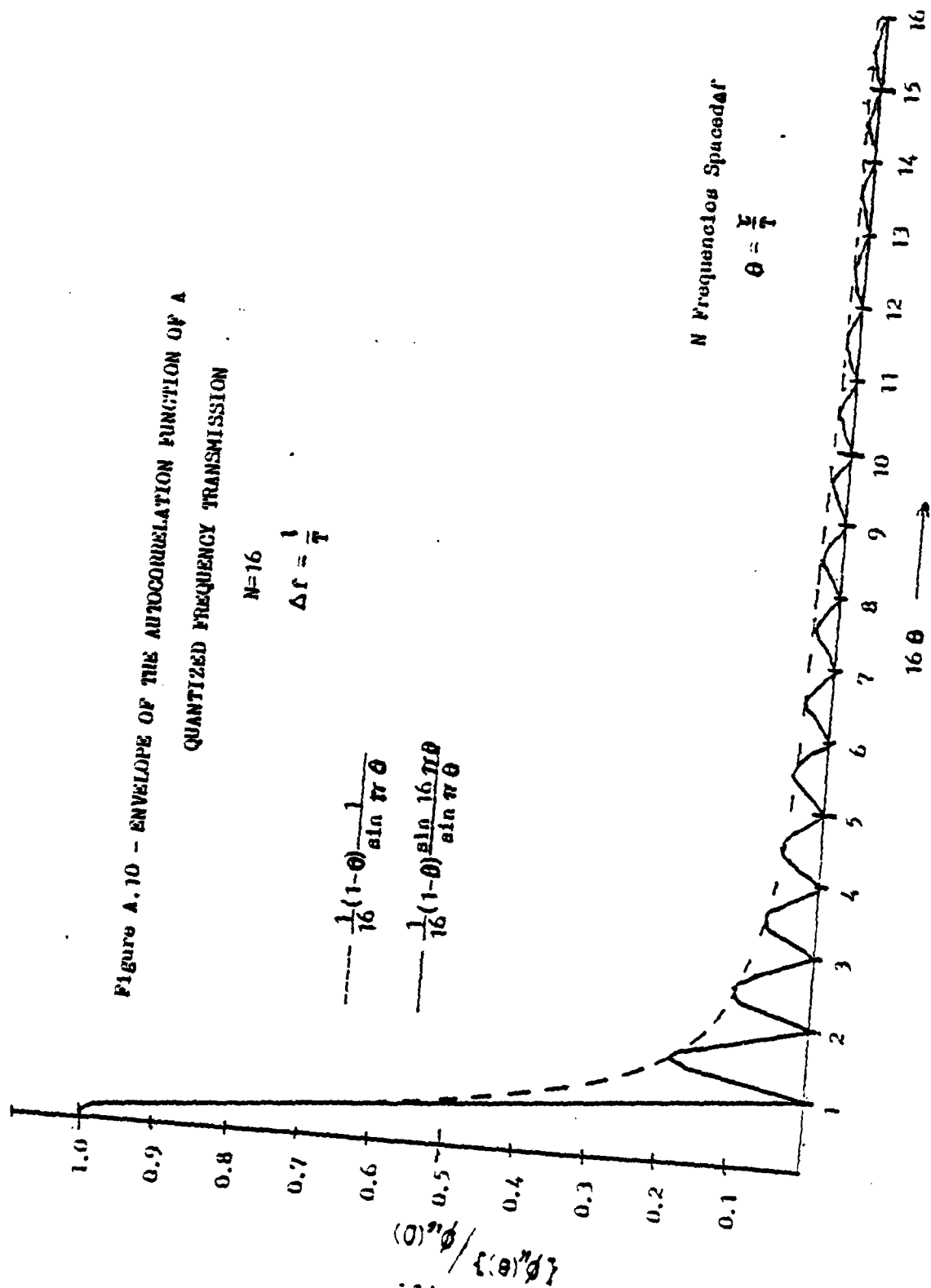
if all the curves are equiprobable $P_n = \frac{1}{2N}$. Now if

$s_n = \sin \Delta wt$, $n = \pm K, \pm(K+1),\ldots, \pm(K+N-1)$

$$\hat{\phi}_s(t_1, t_2) = \frac{1}{2N} \sum_{n=\pm K}^{\pm(K+N-1)} \{\sin n\Delta wt_1 \, \sin n\Delta w \, \Delta wt_2\}$$

$$\hat{\phi}_s(t_1, t_2) = \frac{1}{4N} \sum_{n\pm K}^{\pm(K+N-1)} \{\cos n\Delta w (t_2 - t_1) - \cos n\Delta w (t_2 + t_1)\}$$

and

$$\hat{\phi}_s(\tau, t_1) = \frac{1}{4N} \sum_{n=\pm K}^{\pm(K+N-1)} \{\cos n\Delta w\tau - \cos n\Delta w (2t_1 + \tau)\}$$

$$\frac{1}{2N} \sum_{n=K}^{K+N-1} \{\cos n\Delta w \, \tau - \cos n\Delta w (2t_1 + \tau)\}$$

If the process is stationary t_1 is distributed uniformly over $[0, \tau]$. Therefore, we wish to averate $\hat{\phi}_s(\tau, t_1)$ over t_1. This is done by integrating over the shaded areas of Figure A.11 where $\hat{\phi}_s(t_1, t_2)$ and $\hat{\phi}_s(\tau, t_1)$ possess non-zero value. Hence for $\tau > 0$

$$\phi_s(\tau) = \frac{1}{T} \int_0^{T-\tau} \hat{\phi}_s(\tau, t_1) \, dt_1 \quad ; \quad 0 < \tau < T$$

$$= \frac{1}{2NT} \sum_{n=K}^{K+N-1} \left\{ (T - \tau) \cos n\Delta w\tau + \frac{1}{2nw_0} \sin n\Delta w (2t_1 + \tau) \big]_0^{T-\tau} \right\}$$

$$\phi_s(\tau) = \frac{1}{2N} \sum_{n=K}^{K+N-1} \{(1 - \frac{\tau}{T}) \cos\Delta w\tau - \frac{1}{2nT\Delta w} \sin n\Delta w\tau\} \tag{A.16}$$

Now if $K >> \frac{1}{2T\Delta w}$, the second term may be dropped and

$$\phi_s(\tau) = \frac{1}{2N} (1 - \frac{\tau}{T}) \sum_{n=K}^{K+N-1} \cos n\Delta w\tau \quad ; \quad 0 < \tau < T \tag{A.17}$$
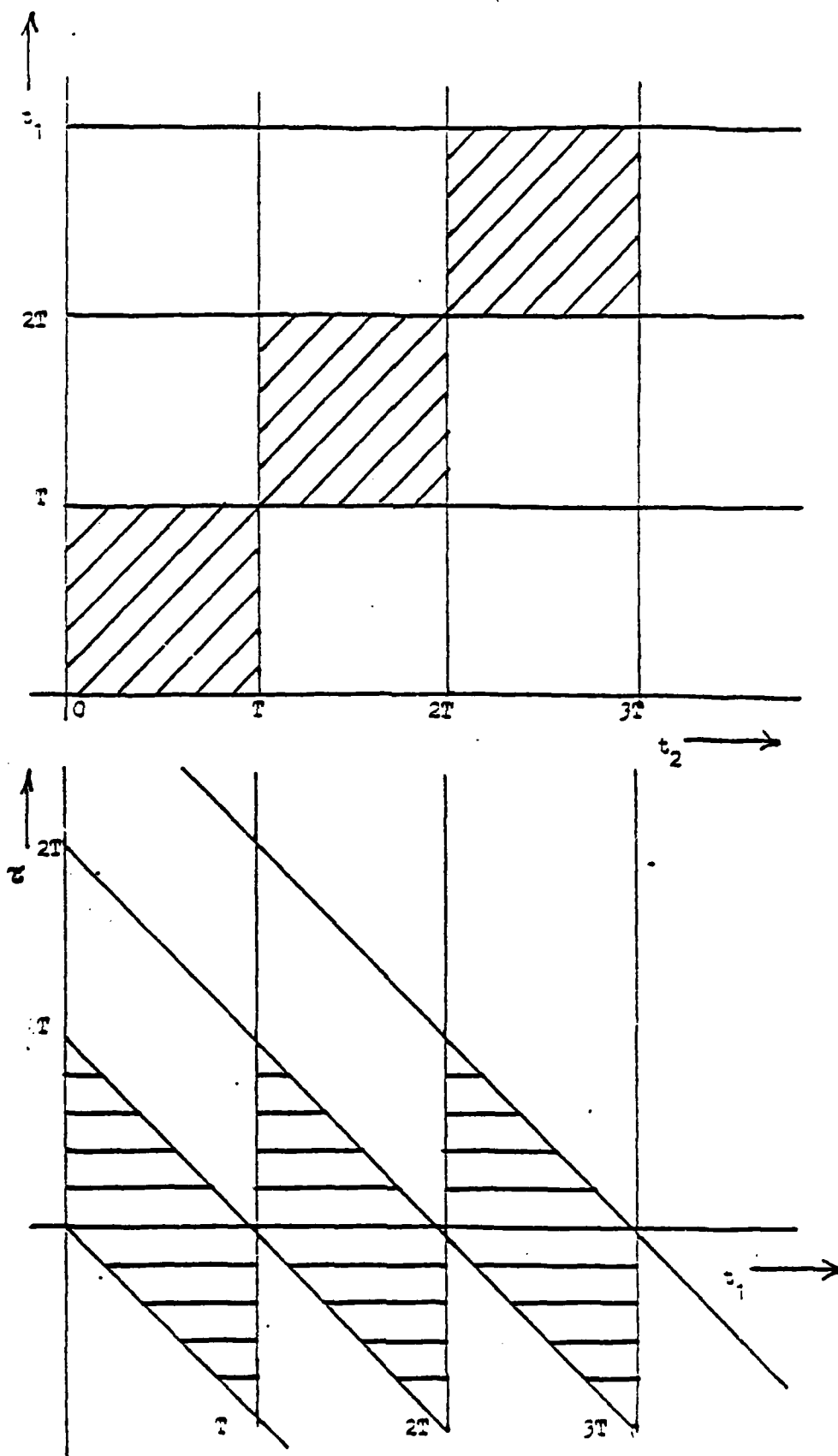
$$= 0 \qquad\qquad ; \quad \tau > T$$

Figure A.11 - Autocorrelation Transformation

Since $\phi_s(\tau)$ is an even function (A.17) is identical with (A.10)

A 7. Correlation Function for Non-coherent FH

For reasons having to do with rapid acquisition, it is common to have the receiver in a FH system operate incoherently. That is, the correlation operation for dehopping is followed by a square law or an envelope detector so that the phase of the r.f. signal is not relevant to the detection process. The consequence of this is that the fine structure of the correlation function is obliterated. A simple calculation can show this phenomenon. The diagram of figure A.12 shows the basic receiver structure where the local FH synthesizer is $s'(t + \tau)$ and is a τ-shifted version of the received signal $s(t)$ <u>and</u> has <u>random</u> <u>phase</u> relative to the received signal. Thus

$$s(t) = \sum_n p^{(n)}(t) \; \sin n \, \Delta wt \quad \text{(as in A.5)}$$

$$s'(t) = \sum_n p^{(n)}(t) \; \sin (n\Delta wt + \Psi_n)$$

Ψ_n are uniform $(0, 2\pi)$; i.i.d.

then the (cross correlation function of s and s' is

$$r_{ss'}(\tau) = \int_{period} s(t) \; s'(t+\tau) dt$$

$$= \frac{1}{N} \left(1 - \frac{|\tau|}{T}\right) \frac{1}{2} \sum_n \cos (n\Delta w\tau + \Psi_n)$$

for $|\tau| < T$ and appears at point A. Next, consider that the non-coherent detector performs a squaring and averaging so that the output (at B) is
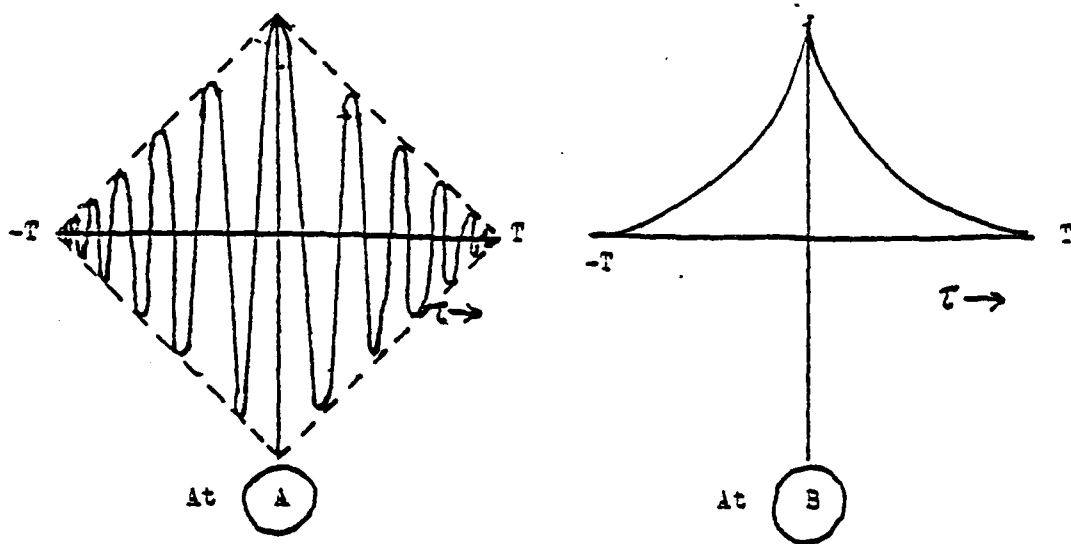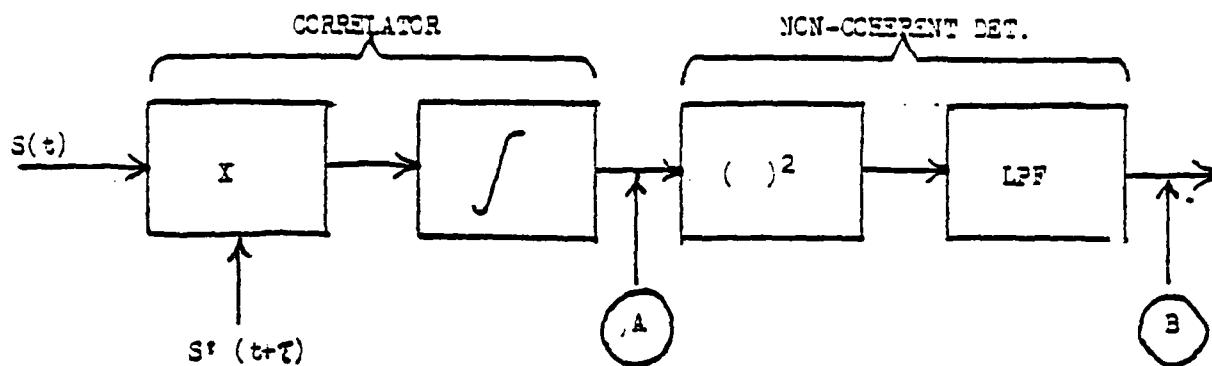
Figure A.12 - Non-coherent correlation

of FH signal

$$\phi_{ss'}(\tau) = E[z^2(\tau)] = [\frac{1}{N}(1-\frac{|\tau|}{T}] \cdot \frac{1}{4} \sum_n \overline{(\frac{1}{2} + \frac{1}{2} \cos(2n\Delta w\tau + 2\Psi_n))}]$$

The overbar indicates expectation or average and is clearly equal to zero so that finally,

$$\phi_{ss'}(\tau) = \frac{1}{8N} (1 - \frac{|\tau|}{T})^2; \quad |\tau| < T$$

as shown. The lack of fine structure allows for much cruder timing accuracy than would otherwise need to be the case for acquisition or tracking. The non-coherent autocorrelation function resembles that of a PN direct sequence with <u>one exception</u>. The reciprocal of the width of the autocorrelation function which is one half the hopping rate is <u>not proportional</u> B_{RF}, the bandspread. Furthermore, the processing gain G_p, determined earlier <u>does</u> <u>not</u> depend on the hopping rate and thus for a given B_{RF} and data rate, R_D, we can achieve any processing gain desired even with arbitrarily low hopping rate! This is easy to see since the processing gain $G_p = \frac{B_{RF}}{B_{DATA}} = \frac{N\Delta f}{\Delta f} = N$ when the data is modulated at 1 symbol/hop (we shall see the evil consequences of this later). Thus G_p is equal only to the number of frequencies used! This is of course appealing since it implies that if,for example,we use N = 1000 frequencies and the jammer doesn't have code tracking ability, his single spot frequency jammer would cause interference only 1 in 1000 thus giving the desired signal a 30dB advantage. The folly of this reasoning is that without coding, that single spot jammer can induce an error rate of 1 in 1000 which may be intolerable. As we will see later, this necessitates the use of coding for FH systems. Without coding however, the achieve-

ment of high processing gain is an illusion as far as jamming protection is concerned.

A.8 Phase Modification for Improved Correlation in Coherent Frequency Hopping

In section A.4 we derived the autocorrelation function for coherent frequency hopping. The autocorrelation function is characterized by many sidelobes which can present great difficulties during sequential acquisition and also during tracking when "tau" modulation or delay locking is used. The problem of sidelobes is avoided by non-coherent correlation as was shown in A.7, but with potential loss in performance, especially in processing gain against a partial band jammer. It is shown here that a modification of the phase structure of the local frequency synthesizer relative to that of the transmitted (or received) signal can be used to eliminate sidelobe effects even when coherent correlation is used. Generalizations are possible.

As in A.4 we write the input frequency hopped waveform as

$$S_1(t) = \sum_{n=K}^{K+N-1} P_T^{(n)} \sin n \, \Delta w \tau \qquad [A.5]$$

The local reference frequency hopping signal is

$$S_2(t) = \sum_{n=K}^{K+N-1} P_T^{(n)} \sin[n\Delta(t+T)+\vartheta_n] \qquad [A.18]$$

When, as before $P_T^{(n)}$ is a random pulse train of 1, 0 and -1 and ϑ_n are a selection of <u>deterministic</u> phase functions which are to be determined to remove the sidelobes in the cross correlation function

- 200 -

$$\phi_s(\tau) = \sum_{n=K}^{K+K-1} (1-\frac{|\tau|}{T})\cos(n\Delta w\tau+\theta_n) \qquad [A.18]$$

The resulting function, of course, reduces to the previous results if $\theta_n=0$ identically for all n. In the interest of writing results relative to the center of the band of hopped frequencies, it is convenient to make the substitution $n-M\pm r$ where r is the harmonic number around the center frequency $M\Delta w$. Then for N even,

$$\phi_s(\tau) = (1-\frac{|\tau|}{T}) \sum_{r=-N/2}^{N/2} \cos[M\Delta w\tau + (r-\frac{1}{2})\Delta w\tau + \theta_r]$$

expanding the cosine and letting $\theta_r=-\theta_{-r}$ we obtain*

$$\phi_s(\tau) = (1-\frac{|\tau|}{T})[\sum_{r=1}^{n/2} \cos\theta_r\cos(r-\frac{1}{2})\Delta w\tau](2\cos M\Delta w\tau) \qquad [A.19]$$

Similarly for N odd

$$\phi_s(\tau) = (1-\frac{|\tau|}{T})[1+2 \sum_{r=1}^{\frac{N-1}{2}} \cos\theta_r\cos r\Delta w\tau](\cos M\Delta w\tau) \qquad [A.20]$$

The last factor in each case, $\cos M\Delta w\tau$, represents the rf center frequency. The other factors are the envelope of the correlation function. When $\theta_r=0$ identically for all r we obtain the result in section A.4. However, if we relate $\cos \theta_r$ to the coefficients of the binomial expansion we can adjust the $\phi_s(\tau)$ function to avoid sidelobes. Thus, let

$$\cos \theta_r = \binom{N-1}{\frac{N}{2}-r}$$

---

*noting that

$$\sum_{r=-n/2}^{N/2} \sin[(r-1/2)\Delta w\tau+\theta_r] = 0$$

- 201 -

and substituting in A.19, we obtain for N even
and

$$\sum_{r=-N/2}^{N/2} \cos[(r-1/2)\Delta w\tau+\theta_r] = 2 \sum_{r=1}^{N/2} \cos \theta_r \cos(r-1/2)\Delta w\tau$$

The envelope of the correlation function is

$$\text{Env } \emptyset_s(\tau) = (1-\frac{|\tau|}{T}) \sum_{r=1}^{N/2} \binom{N-1}{N/2-r} \cos(r-1/2)\Delta w\tau \qquad [A.21]$$

This series may be evaluated in closed form by examining the binomial expansion

$$\cos^{N-1} \frac{\Delta w\tau}{2} = (\frac{e^{i\Delta w\tau/2}+e^{-i\Delta w\tau/2}}{2})^{N-1}$$

$$= 2^{1-N} \sum_{K=0}^{N-1} \binom{N-1}{K} e^{iK\Delta w\tau/2} \cdot e^{i(N-1-K)\Delta w\tau/2}$$

$$= 2^{1-N} \sum_{K=0}^{N-1} \binom{N-1}{K} e^{i(N-1-2K)\Delta w\tau/2}$$

since N is even this can be written

$$= 2^{1-N} \sum_{K=0}^{N/2-1} \binom{N-1}{K} e^{i(N-1-2K)\Delta w\tau/2}$$

$$= 2^{1-N} \sum_{K=0}^{N/2-1} \binom{N-1}{N-1-K} e^{-i(N-1-2K)\Delta w\tau/2}$$

where we have changed the index of summation in the second series from K to N-1-K. Now since $\binom{N-1}{K} = \binom{N-1}{N-1-K}$ we have

$$= 2^{2-N} \sum_{K=0}^{N/2-1} \binom{N-1}{K} \cos(N-1-2K)\Delta w\tau/2$$

letting $r = N/2-K$

$$= 2^{2-N} \sum_{K=0}^{N/2-1} \binom{N-1}{N/2-r} \cos(r-1/2)\Delta w\tau \qquad [A.22]$$

Equation A.22 contains the summation in the envelope of the correlation function, and we can finally write for N even

$$\text{Env } \emptyset_s(\tau) = (1-\left|\frac{\tau}{T}\right|) \cdot 2^{N-2} \cdot \cos^{N-1} \Delta w\tau/2 \qquad [A.23]$$

and for N odd,

$$\cos^{N-1} \Delta w\tau/2 = 2^{1-N}(1+2 \sum_{K=1}^{N/2-1} \binom{N-1}{\frac{N-1}{2}-K} \cos K\Delta w\tau)$$

and we would choose (see equation A.20)

$$\cos \theta_r = \binom{N-1}{N-1/2-r}$$

In either case the envelope of the correlation function given by A.23 in the range $|\tau| \leq T$ and zero elsewhere is strictly a monotonic function in this range with no sidelobes. A normalized plot of this function for N=16 and T=0.9/Δf is shown in figure

A.13. The required relative phases of the hopping frequencies is shown in figure A.14.

The implementation of such a correlation requires that the receiver frequency synthesizer adjust the relative phase of the selection frequencies such that the rth frequency from the center of the band has a relative phase

$$\theta_r = \cos^{-1}\left(\frac{N-1}{N/2-r}\right) \qquad [A.24]$$

for N even and

$$\theta_r = \cos^{-1}\left(\frac{N-1}{N-1/2-r}\right) \qquad [A.25]$$

for N odd.

Actually, the binomial phase law[A.24] and [A.25] is similar to the technique used in phased antenna arrays to eliminate sidelobes. Extensions to this technique are possible. If the amplitude of the individual frequencies that are used for local correlation are also amenable to variation, a possible advantage here is the prospects of obtaining very narrow auto-correlation functions for frequency hopping which can then be used for accurate timing and/or ranging.

A.9 Partial Band Jamming

It was established previously that although the processing gain of a frequency hopping (FH) system is N, the number of frequencies used, that the probability of error or bit error rate (BER) is 1/N if we transmit at the rate of 1 bit per hop and the jammer transmits on one frequency. This is an extreme case of partial band jamming since the jammer needs only to concen-

Figure A.13 Normalized Correlation Function for Frequency Hopping with Binomial Phased Modification.

Figure A.14 Binomial Phase Modification

Harmonic Number

Phase Shift (degrees)
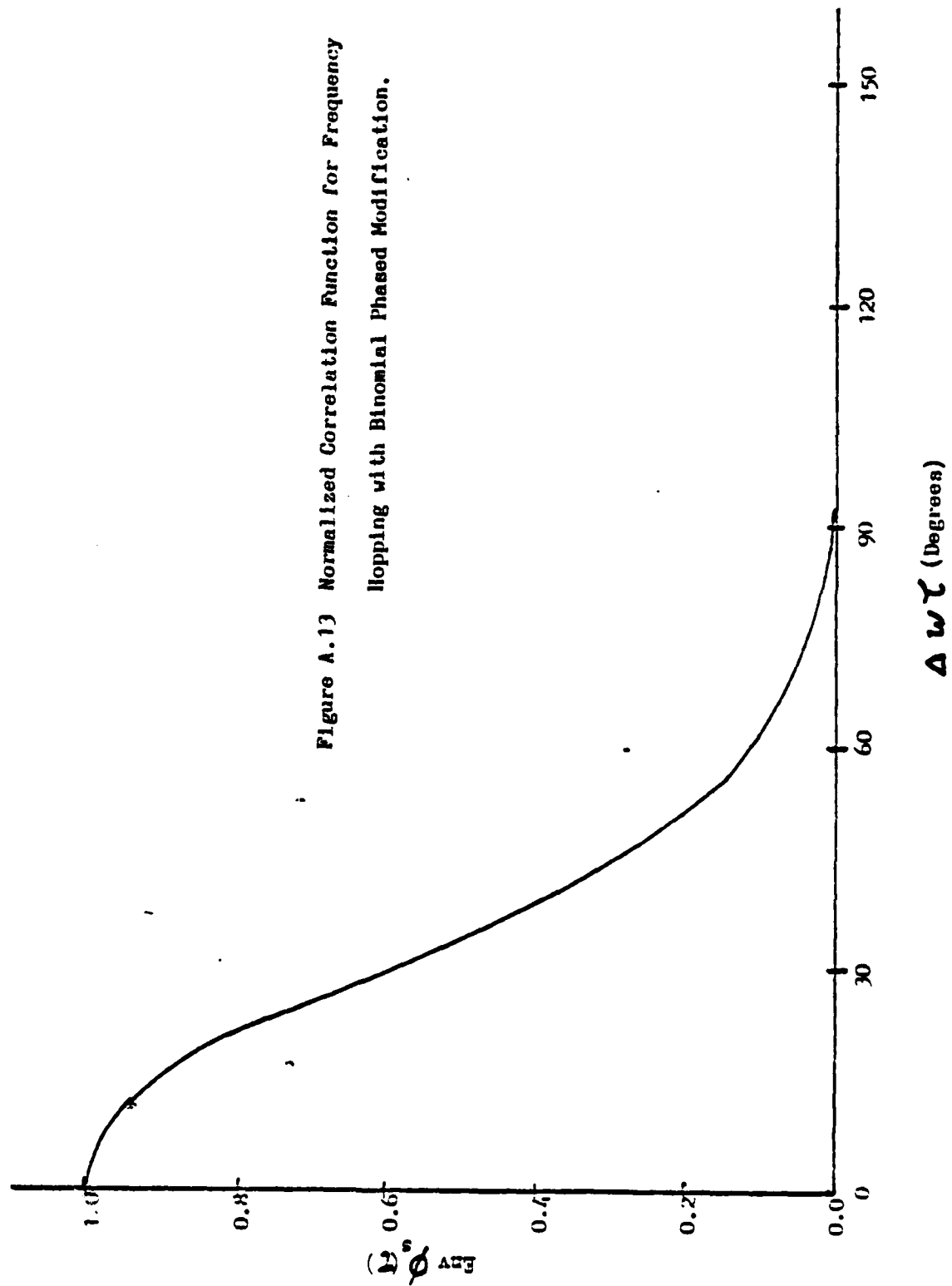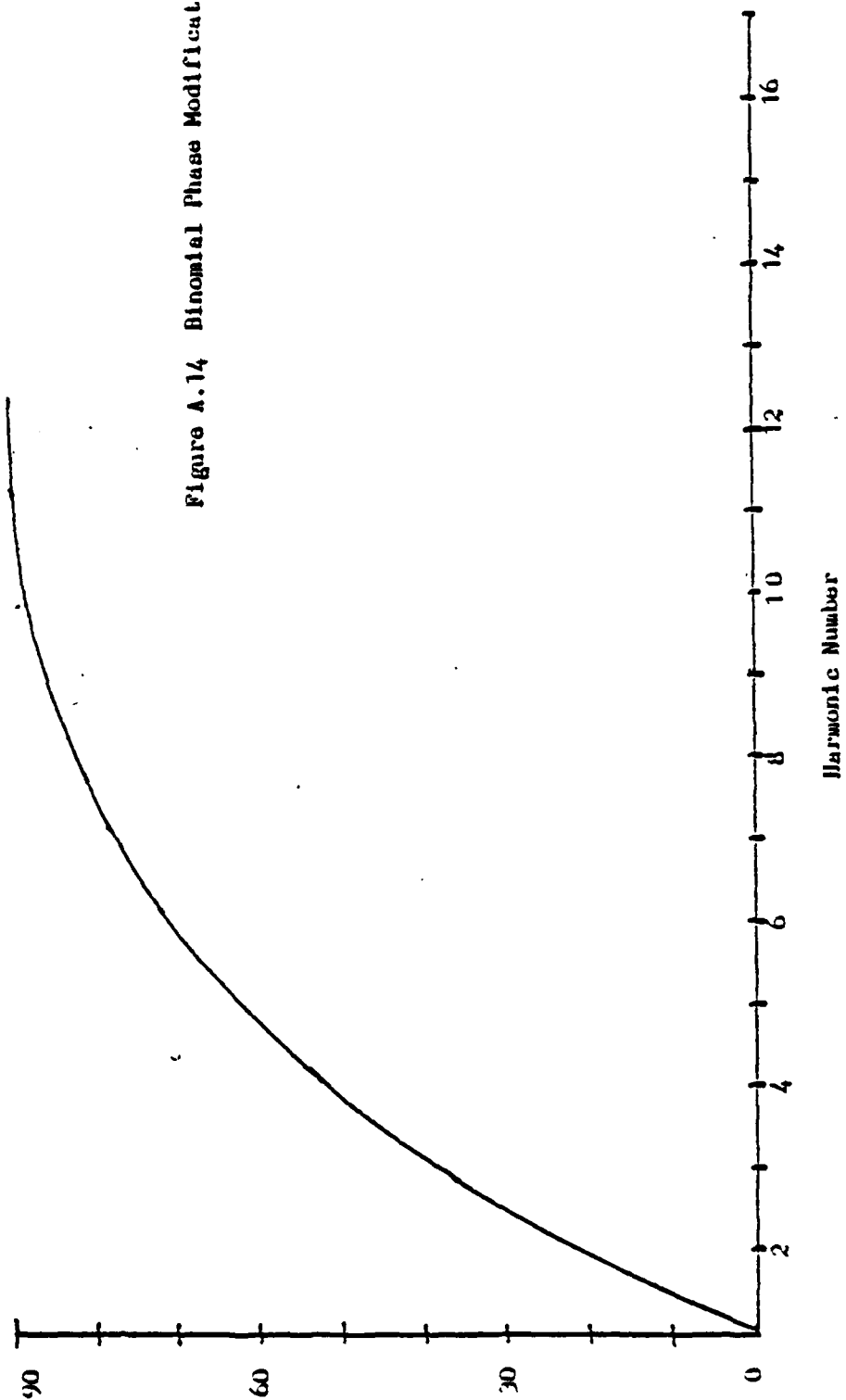
trate all his power at one frequency. A more extreme case
ensues if the frequency hopper uses non-coherent FSK signalling
with deterministic or adjacent mark and space frequencies. Under
these circumstances the jammer need only detect the instantaneous
frequency and transmit the <u>other</u> frequency. Thus if a mark
is sensed then a space is transmitted and vice versa. Indeed this
regimen will degrade the BER to a totally useless 1/2. This
illustration points out the need <u>at least</u> to pseudo-randomize
the mark-space frequencies. This is especially the case under
slow hopping where a potential repeater jammer has ample time
to measure the received frequency and take appropriate action.

In order to fully appreciate the effects of partial band
jamming we will consider several modulation schemes incorporated
in F.H. systems and the effects of a band of tone jammers.
To begin, suppose we have uncoded binary coherent phase shift
keying (PSK) and the signal is

$$s(t) = \pm \sqrt{\frac{2E_b}{T}} \cos(w_o t) \qquad [A.26]$$

Then a jamming frequency will cause a bit error if it is:

1. on frequency w_o

2. of phase and magnitude to cause a polarity change
   in the output of the demodulation.

The jamming signal, if censored on w_o is:

$$j(t) = \sqrt{\frac{2E_J}{K}} \cos(w_o t + \theta)$$

where E_J is the total jamming energy/bit spread among K carriers,
and θ is uniformly distributed bewteen 0 and 2π. The probability
of a bit error is therefore equal to

$$P(\epsilon) = P(\text{jammer produces error} \mid \text{jammer on frequency})$$

$$\cdot \; P(\text{jammer on frequency})$$

$$= P\left(-\sqrt{E_b} \le \sqrt{\frac{E_J}{K}} \cos \theta \le 0\right) \cdot \frac{K}{N}$$

$$P(\varepsilon) \quad = \quad \begin{cases} \dfrac{K}{\pi N} \cos^{-1} \sqrt{\dfrac{KE_b}{J_o N}} \; ; & \dfrac{E_b}{J_o} < \dfrac{N}{K} \\[4mm] \\ 0 & \dfrac{E_b}{J_o} \ge \dfrac{N}{K} \end{cases} \qquad \text{[A.27]}$$

Where it is assumed that the jammer phase is uniformly distributed $(0,2\pi)$ and where $J_o \triangleq E_J/N$ is the jammer density since it measures the jammer power at each frequency if the total jammer power were equally spread amongst each of the N hopping frequencies. From A.27, it is apparent that their exists a K/N which maximized $P(\varepsilon)$. Treating K/N as a continuous variable for simplicity allows us to find the maximum. We can differentiate and set the result equal to zero when the maximum falls inside the boundaries $1/N < 1$. Whether the maximum falls in this range depends on E_b/J_o as depicted in figure A.15. When E_b/J_o is very small the $P(\varepsilon)$ versus K/K cause is dominated by $K/\pi N$ and it is seen that the maximum within the boundary occurs at $K/N = 1$. The actual maximum value of E_b/J_o for which this condition holds is $E_b/J_o = 0.64$ (-1.9dB). Beyond that range the peak $P(\varepsilon)$ occurs inside the boundaries until $E_b/N_o = 0.64N$. The resulting maximum is then given by
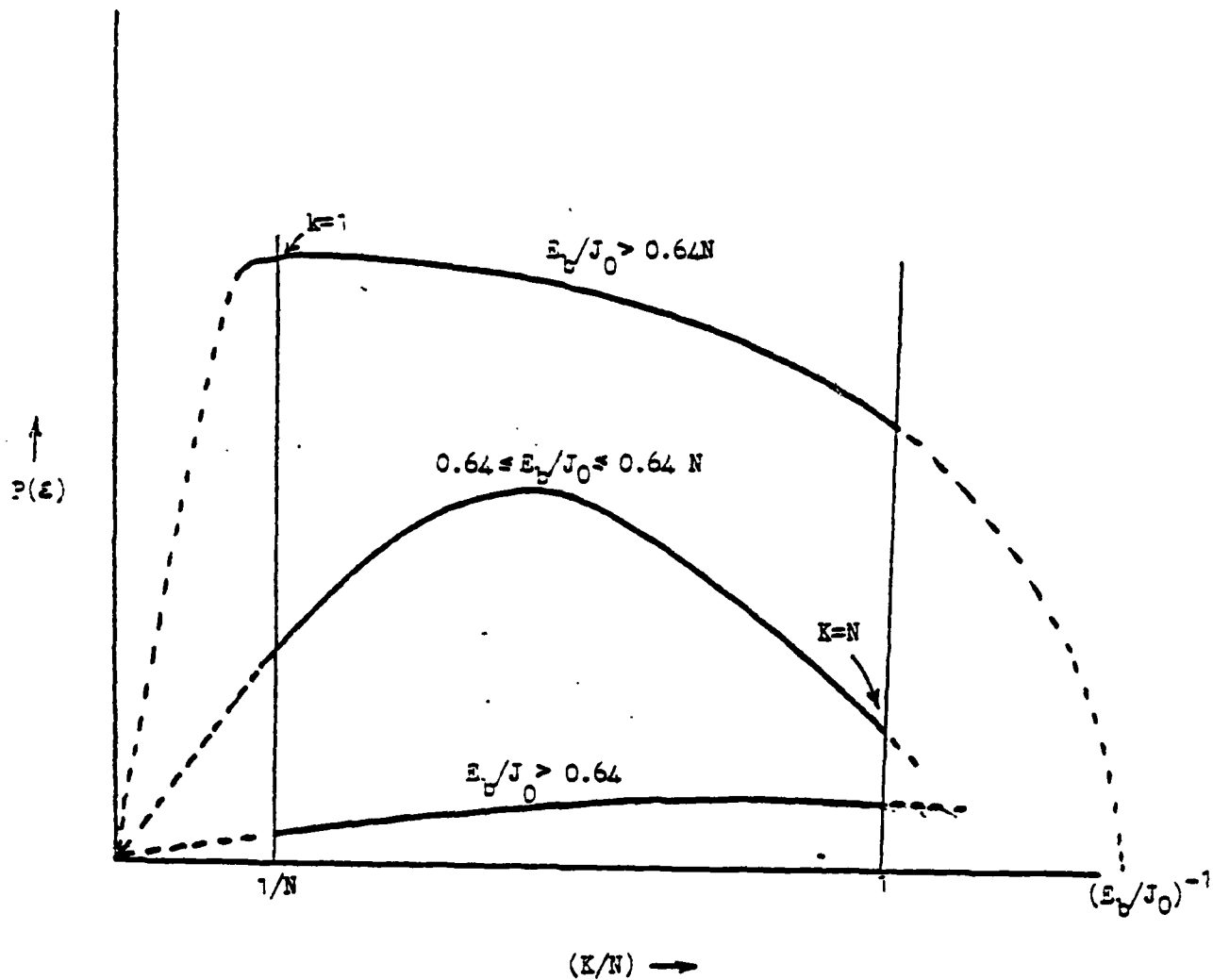
Figure A.15 Optimal (MAX P(ε)) Partial Band Jamming

$$\max_{0 \le K/N \le 1} P(\varepsilon) \begin{cases} 1/\pi \, \cos^{-1} \sqrt{E_b/J_o} \; ; \; E_b/J_o \le 0.64; \; K^* = N \\[2em] \dfrac{0.13}{E_b/J_o} \; ; \; 0.64 < E_b/J_o \le 0.64N; \; 1 < K^* < N \\[2em] 1/\pi \, \cos^{-1} \sqrt{E_b/NJ_o} \; ; \; E_b/J_o > 0.64N; \; K^* = N \end{cases}$$

Where K^* is the optimum number of jamming frequencies to do the most damage i.e. maximize $P(\varepsilon)$. To illustrate, suppose $N=1000$ frequencies, then for $E_b/J_o < 640$ (=28.1dB) there exists a strategy for the jammer which does not require him to spread his power over the entire band of 1000 frequencies. If, for example, $E_b/J_o = 10$ (10dB) then such a strategy can induce a $P(\varepsilon) = 0.013$ which is unacceptable. Recall that for $E_b/N_o = 10$ in Gaussian noise, $P(\varepsilon) < 10^{-3}$. All of the above discussion was for uncoded, coherent binary phase shift keying with random frequency hopping for ECCM, and discrete, partial band tone jammers. Coherent BPSK was used to illustrate and highlight the mechanism in a way which is transparent and easy to comprehend. The same analysis can be carried out for non-coherent nonbinary, FSK, DPSK, etc. and for partial band noise jamming as well with the same dismal results.

Since most frequency hopping systems do employ non-coherent detection, a simple calculation with partial band noise jamming is useful in highlighting the phenomenon at work. For non-coherent FSK in full band Gaussian noise with power spectral density N_o watts/HZ, the minimum probability of error receiver achieves

$$P(\varepsilon) = \frac{1}{2} \exp \left(-E_b/2N_o \right)$$

- 210 -

Now if a two level[1] additive noise has power density

$$N_0/p \text{ in a fraction } 0 \le p \le 1 \text{ of the band}$$

and

$$0 \text{ in a fraction } 1-p \text{ of the band}$$

Then a random frequency hopper will encounter this noise density with probability p and will be (virtually) noiseless or error-free with probability 1-p. Thus for this partial band jammer

$$P(\varepsilon) = p/2 \exp-(p \, E_b/2N_0)$$

(p here plays the same role as K/N did in the partial band tone jammer above). It is apparent that the p which maximizs $P(\varepsilon)$ is $p = 2N_0/E_b \le 1$ or

$$\max_{0 \le p \le 1} P(\varepsilon) \ge \frac{1}{e} \frac{1}{E_b/N_0} = \frac{0.37}{E_b/N_0}$$

With equality when $E_b/N_0 = 2$ in which case full band (p=1) jamming is required to induce the given probability of error. As an example, again take $E_b/N_0 = 13.7$dB which, recall results in $P(\varepsilon) < 10^{-5}$ for full band Gaussian noise for noncoherent FSK. A partial band jammer occupying a fraction p=0.2 of the band (optimal) will cause the error rate to be 0.016 which is totally unacceptable. A different way to look at this issue which is even more dramatic is to ask what E_b/N_0 is, in fact, required with the worst case partial band jammer to achieve 10^{-5} error rate?

---

[1] It can be shown that there is no worse distribution than two level.

The answer is 45.68dB! The use of multiple level signalling actually makes matters worse since if we employed M-ary FSK, the results for $P(\varepsilon)$ would be multiplied roughly by $(M-1)/\log_2 M$.

The basic behavior in either case (coherent PSK, non-coherent FSK) is the same. The error rate reduces only inversely with E_b/N_o rather than exponentially as is the usual situation with wideband noise. This phenomenon is characteristic not only of partial band jamming but of Rayleigh fading as well since the effects of the two are very much related. With this inverse behavior signalling can be virtually distrupted with quite little power expenditure on the part of a jammer. The only solution is to devise a modification of the data encoding on the frequency hopper which would force the jammer to spread and so that he will achieve no advantage from partial band jamming. Two techniques are available. One is diversity. I.e. send data in a multiple such as on different frequencies. In frequency hopping this is readily achieved by having a bit sent over the duration of many frequency hopping "chips". This is known as bit-splitting and it has the effect of preventing only a few spot jammers from obliterating any bit but it reduces the data rate accordingly. The second method is coding. This too requires effective bit-splitting but if the codes chosen are sufficiently powerful then it may be possible to achieve the desired performance without sacrificing the data rate. Indeed diversity itself is a repetition code which is not optimum in this sense.

A.10 Coding for Frequency Hopping

Because partial band jamming can be so effective against
a frequency hopper, it is natural to inquire whether channel
coding can help to mitigate the hazard. It will be shown that
even simple codes such as repetition codes will work, but more
importantly, that with coding it is possible to achieve not
only the full practical processing gain expected of frequency
hopping but that a coding gain can be realized on top of everything.

To illustrate the ideas succinctly we will use a simple
odd-repetition binary code to show how partial band jamming
strategies can be made ineffectual. Consider a "slow hopping"
transmitter that takes 3 hops to present one information bit.
Then a noncoherent dehopping receiver can use a majority vote
decision rule to decide a bit. That is, the receiver announces
a binary "1" if two or more "1"s appear in a triplet of three
hops. Since the three hops during one bit are pseudo random, a
single spot jammer with power equal to that of the received signal
cannot cause an information bit error since this jammer will at most
wipe out or cause an error in one out of three chips making up a
bit and majority logic decision will always be correct, providing
that the frequency triplet is distinct. It, in fact, now requires
at least two spot jammers to cause a possible error, (if the two
spot jamming frequencies happen to coincide with 2 of the 3 hop
frequencies and to cancel them, due to fortuitous phase relationships.)

For example, if there were $N=1000$ random frequencies and
one jammer, the probability of a bit error is then $P_e = \binom{3}{2} p^2(1-p)$
$+ \binom{3}{3} p^3 = 3 \times 10^{-6}$ where $p = 10^{-3}$ is the probability of a spot

match. This is to be compared with $P_e = 10^{-3}$ if we had no repetition code. This result, which gives close to three orders of magnitude improvement is not surprising since the data rate has been reduced by $\frac{1}{3}$. Therefore, in terms of processing gain:

$$P.G. = \frac{B_{RF}}{B_{DATA}} = \frac{B_{RF}}{R} \text{ and if } R' = \frac{1}{3} R; \ P.G.' = 3 \ P.G. \text{ This is a}$$

4.7 dB increase. If there is bandwidth to "burn" then repetition codes do offer protection for a frequency hopper against partial-band jamming. A design example will illustrate.

Design Example

> Let R = Binary data rate
>
> B_{RF} = Radio frequency (one-sided) bandwidth
>
> N = Number of frequency hop channels
>
> T = Hop rate
>
> M = Number of hops per bit
>
> J = Number of spot jammers

It is assumed that each of the spot jammers is of sufficient power to cause a chip error if the frequencies collide.

Then $P_e = \binom{M}{[M/2]} p^{[M/2]}$ where $p = J/N$ and $[x]$ = smallest integer $\geq x$. Then since orthogonality requires that $N = B_{RF}T$ and
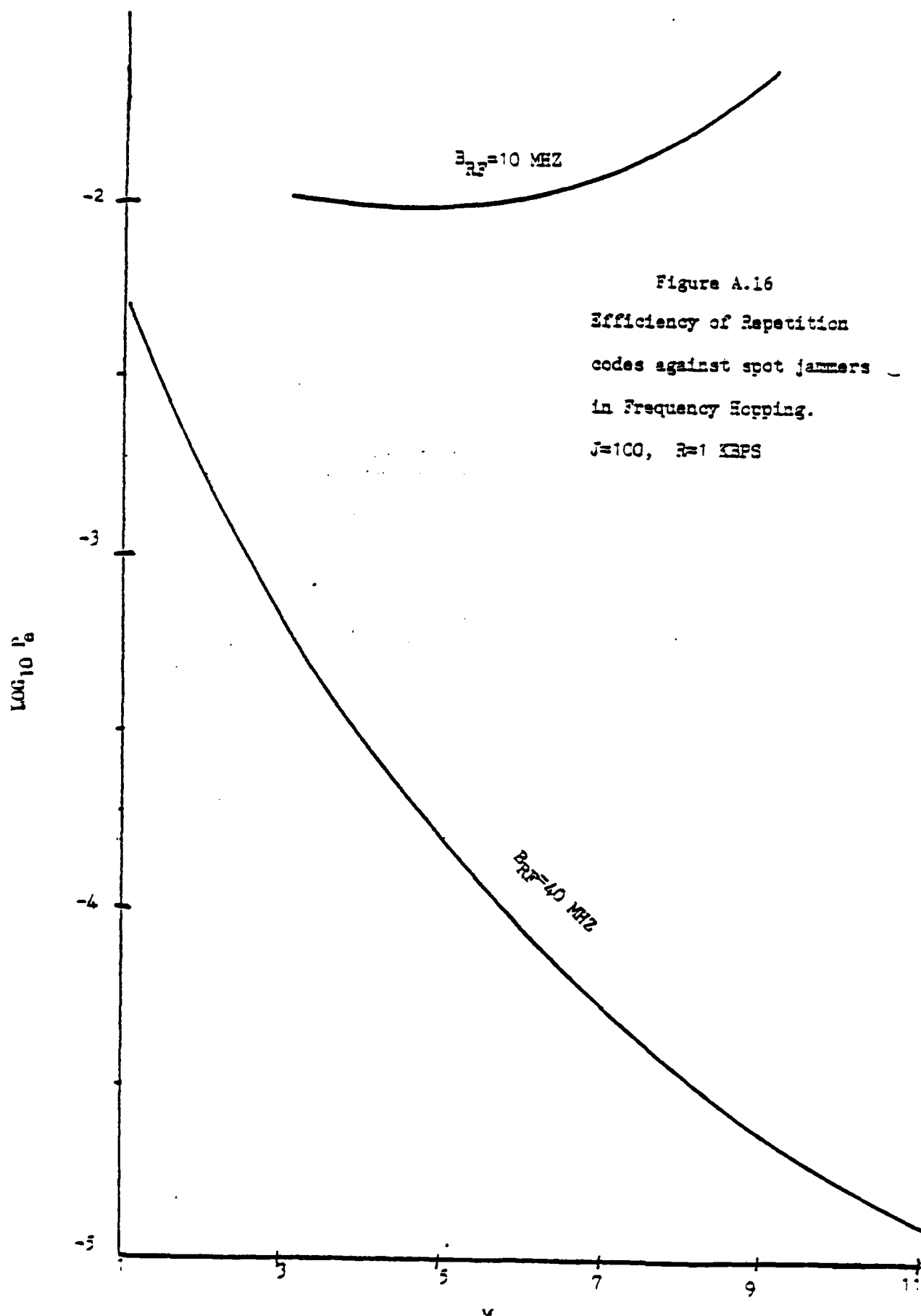
$R = \frac{1}{MT}$, combining terms we get:

$$p = \frac{J}{N} = \frac{JMR}{B_{RF}} \tag{A.27}$$

$$\text{and} \qquad P_e = \binom{M}{[M/2]} p^{[M/2]} \tag{A.28}$$

- 211 -

We usually specify P_e, R, and B and we need to solve for M necessary to deny the jammer a partial band advantage. Instead we will illustrate the behavior with M of P_e with the B_{RF} as a parameter. This is illustrated in figure A.16 for B_{RF} = 10MHz and 40 MHz. The minimum at 10 MHz occurs at about M = 5 while at 40 MHz P_e is minimized at about M = 11. The information rate is fixed at 1KBPS and the number of jammers is assumed to be 100. With even greater bandwidth a lower P_e is achievable but the data rate remains the same. The problem is that this repetition code is very wasteful of the bandwidth since similar reductions can be obtained by the use of a more sophisticated code.

To achieve a large distance between codewords with more bits/chips, (more than 1/M as in repetition codes) it is advantageous to use a multialphabet code. Multialphabet codes are natural for frequency hopping since it is straightforward to identify each of the alphabet symbols by a particular frequency. For q-ary alphabet with q = p^S (a prime raised to any integer power) the class of <u>Reed-Solomon</u> (R-S) codes can be constructed with (n,k) = (q-1, q-d). n is the number of channel digits, k is the number of information digits, q is the alphabet size, and d is the minimum distance, such that the codewords have the greatest possible minimum distance for a specified (n, k). In this sense the R-S codes are optimal. For example if q = 2^3 = 8; d = 6; (n, k) = (7, 2) then a codeword consists of 7 octal digits, two of them being information digits. Thus a seven digit word contains 6 bits and there are therefore 2^6 = 8^2 = 64 codewords. Since the

Figure A.16
Efficiency of Repetition
codes against spot jammers
in Frequency Hopping.
J=100, R=1 KBPS

distance is 6, a jammer must cause at least four errors to result in an undetected symbol error. A repetition code would have required 9 chips for the same performance. This is a savings of 2/7 or 28%.

The R-S codes have the further property of 1) being almost orthogonal, 2) being easily encodable, and 3) having a sufficient algebraic structure to allow for a variety of decoding algorithms. The R-S codes are constructed using a finite 2^n or $GF(2^n)$ arithmetic. For example for the arithmetic of $2^3 = 8$ elements the multiplication and addition table is shown in figure A.17. The encoder is a mapping of octal doublets (6 bits) into octal septuplets. The seven elements of the octal codework C_1, C_2, C_3, C_4, C_5, C_6, C_7 are obtained from the two data elements α, β by the equations

$$C_1 = \alpha + \beta \qquad\qquad C_5 = \alpha + 6\beta$$
$$C_2 = \alpha + 4\beta \qquad\qquad C_6 = \alpha + 7\beta$$
$$C_3 = \alpha + 2\beta \qquad\qquad C_7 = \alpha + 3\beta$$
$$C_4 = \alpha + 5\beta$$

where α, β are elements of $GF(2)$ and satisfy the addition and multiplication table of figure A.17. The implementation using binary registers is shown in figure A.18. Reed-Solomon and other nonbinary codes are ideally suited to being implemented in a frequency hopping system since the alphabet symbols can be represented by different frequencies. For example, in figure A.18, the eight digits are represented by eight offset frequencies separated by 2Δ Hertz. Thus the operation of the encoder in this case and shown in figure A.18 proceeds from capturing a block

Figure A.17

R-S CODE ARITHMETIC ON OCTAL DIGITS GF(8)

X	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7
2	0	2	6	4	5	7	3	1
3	0	3	4	7	1	2	5	6
4	0	4	5	1	2	6	7	3
5	0	5	7	2	6	3	1	4
6	0	6	3	5	7	1	4	2
7	0	7	1	6	3	4	2	5

+	0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6	7
1	1	0	3	2	5	4	7	6
2	2	3	0	1	6	7	4	5
3	3	2	1	0	7	6	5	4
4	4	5	6	7	0	1	2	3
5	5	4	7	6	1	0	3	2
6	6	7	4	5	2	3	0	1
7	7	6	5	4	3	2	1	0

Figure A.18

R-S ENCODER

$$C_1 = \alpha + \beta \qquad\qquad C_5 = \alpha + 6\beta$$

$$C_2 = \alpha + 4\beta \qquad\qquad C_6 = \alpha + 7\beta$$

$$C_3 = \alpha + 2\beta \qquad\qquad C_7^7 = \alpha + 3\beta$$

$$C_4 = \alpha + 5\beta \qquad\qquad \alpha, \beta = GF(8)$$

C_1	C_2	C_3	C_4	C_5	C_6	C_7
0	1	0	0	0	1	1
0	1	0	1	1	1	0
1	1	0	1	0	0	0
1	7	0	3	2	6	4

α: 1 0 1

β: 1 0 0

CODEWORD
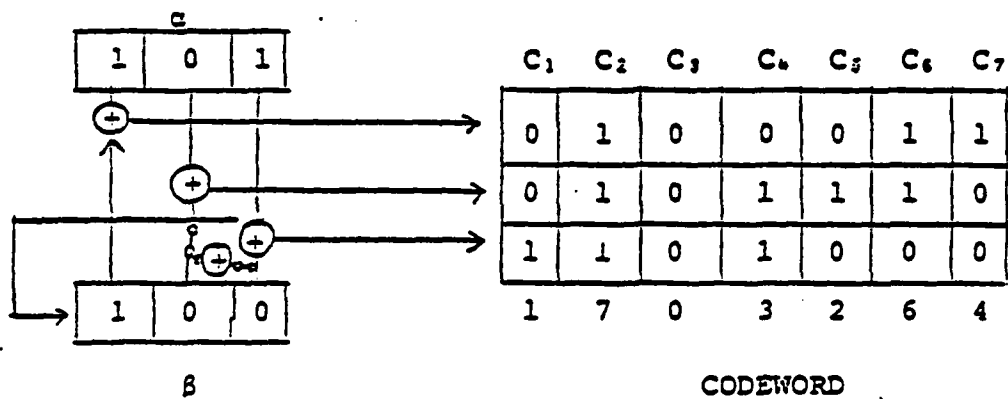
DIGITS	FREQ $(f_c +)$
7	7Δ
6	5Δ
5	3Δ
4	$+\Delta$
3	$-\Delta$
2	-3Δ
1	-5Δ
0	-7Δ

of 6 data bits (2 octal digits) given as α, β = 101, 100 and
yielding seven octal digits 1703264. These later digits in turn
induce a frequency hop shown as a typical word in figure A.19.
The numbers and the transmission format are those used in TATS
(Tactical Transmission System). A block diagram of the transmitter
in figure A.20 shows not only the R-S frequency hop code
but the major frequency hopping which can be invoked when A/J
conditions required it.

Figure A.21 shows the receiver-decoder. A bank of matched
filters is used to detect the R-S symbols (incoherently). The
maximum output is quantized (for soft decisions) and is used to
establish a received word. The decoding can be accomplished
algebraically using the Berlekamp algorithm if hard decisions
are used. The decoder may require the ability to do arithmetic
over (GF(8) and to interactively solve simultaneous equations
over that field.

A.11 Digitally Controlled Frequency Synthesis

The fundamental component in a frequency hopping system is
a synthesizer which has the following properties:
- Fast hopping rate (of the order of 1µsec or less)
- Low spurious response
- Fast settling time
- Digital control

All of these requirements can be met with a direct frequency
synthesis technique which can be modularized into a sequence of
(frequency) divide and (base frequency) add operations. The
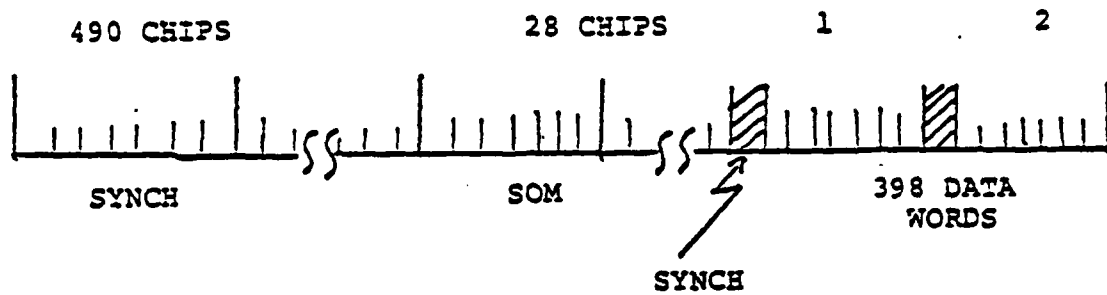
Figure A.19a

TRANSMISSION FORMAT (7 digit R-S)

490 CHIPS   28 CHIPS   1   2

SYNCH   SOM   398 DATA WORDS

SYNCH

Figure A.19b

TYPICAL WORD (MINOR HOP PATTERN)

$T_C = 10\text{ms}$

$\Delta = 50\text{HZ}$

$BW = 160 = 800$

$R = \dfrac{66}{80\text{ms}} = 75\text{bps}$

$T_C = 312.5\text{ms}$

$\Delta = 1.6\text{KHZ}$

$BW = 25.6\text{ KHZ}$
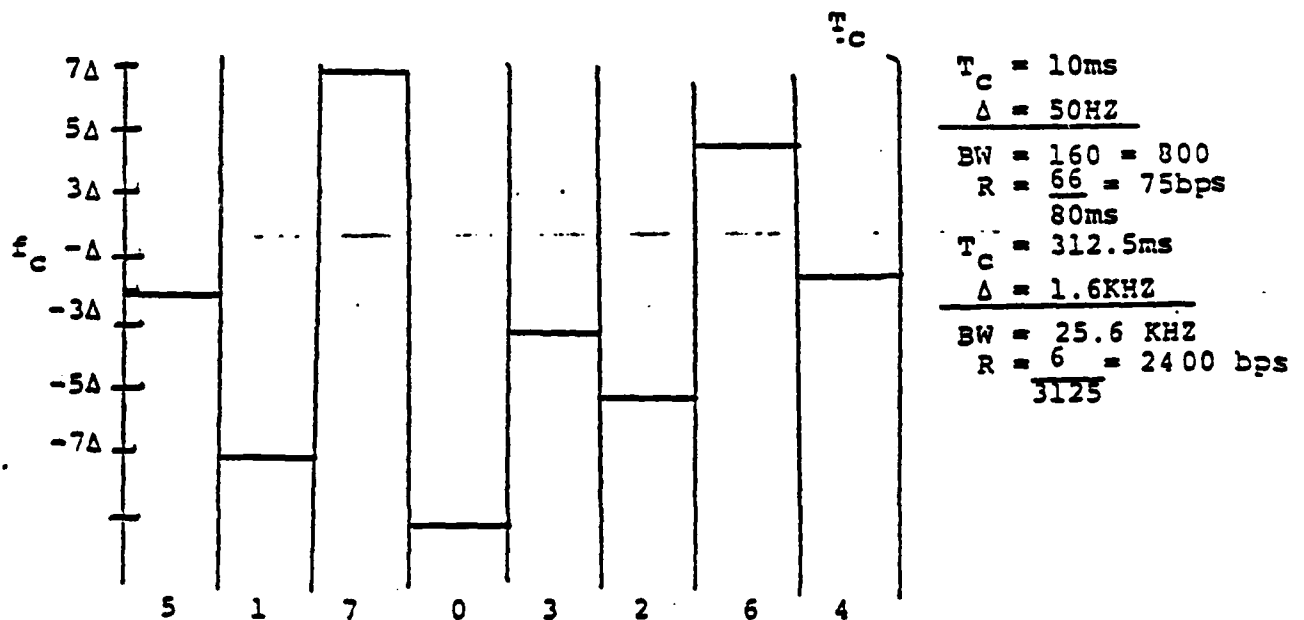
$R = \dfrac{6}{3125} = 2400\text{ bps}$

Figure A.20

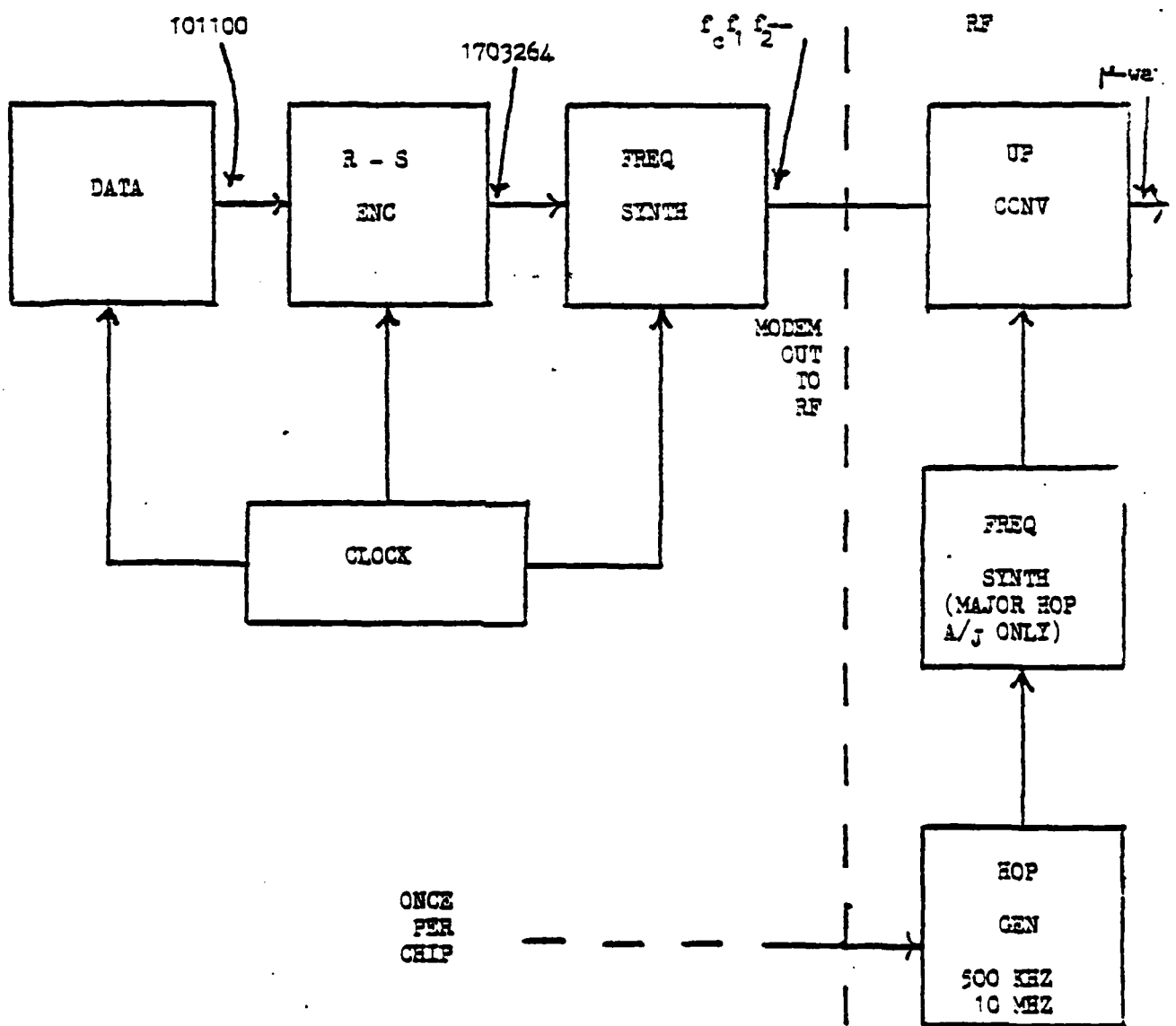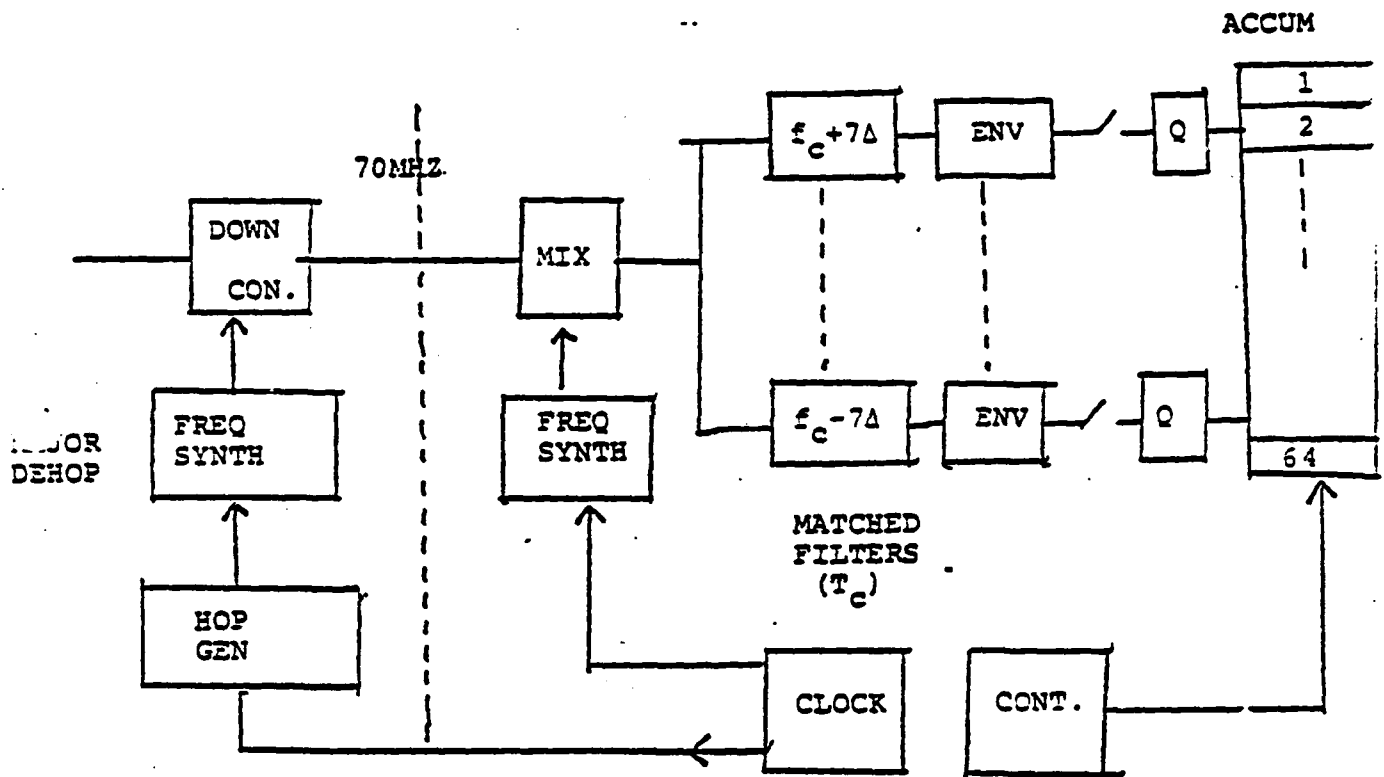## Frequency Hopping Transmitter with R-S Encoding

Figure A.21
RECEIVER/DECODER

basic idea is illustrated in figure A.22. Here we see two stages
of a possible S stage system. The K base frequencies and the
K^{th} subharmonic of the center frequency, f_c may all be derived
from a single master oscillator. At each stage there is a
mixer-band pass filter which serves to derive the sum frequency
of H input followed by a divide by K frequency divider. The
divider is omitted in the last stage. A gate selects which one
of the K base frequencies is chosen at each stage. Thus the
control bit at each stage consists of $\log_2 K$ binary digits.

An example here illustrates the operation of the synthesizer.
We wish to construct a selection of one of eight frequencies,
spaced by 1 (unit frequency) and having a band center of 10 ($f_c =$
10). Figure A.23 defines the problem and shows that the two
base frequencies $f_1 = 3$ and $f_2 = 7$ are required if 3 stages are
used ($2^3 = 8$). The details are further illustrated in figure A.24
where we follow the outputs of stages 1, 2, and 3 respectively to
finally achieve the objective. Notice that the control in this
instance is 3 bits, one per stage. Had we employed say 10
stages with one bit each we would have achieved $2^{10} = 1024$
frequencies. Alternatively we could have 2 bits per stage and 5
stages for the same result. In that case four base frequencies
would be necessary but faster settling time would be achieved
since the signal would have to propagate through half as many
filters and frequency dividers.

We summarize by listing the advantages and disadvantages
of digitally controlled synthesizers currently available.
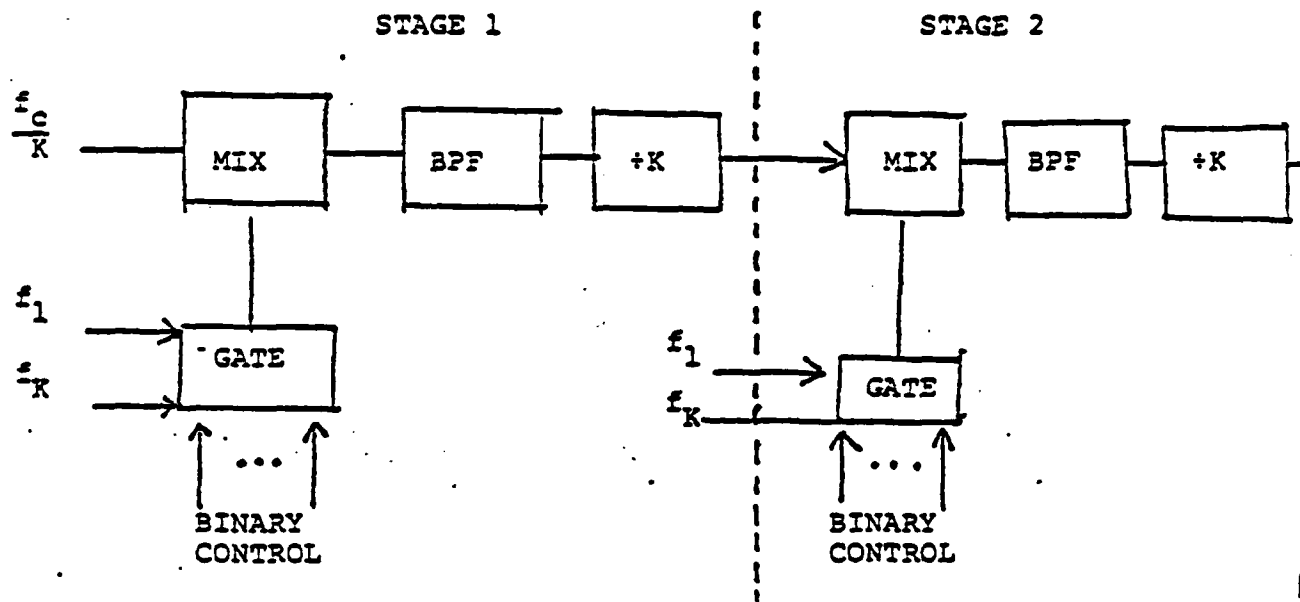
K base freq spaced δf and symmetrical about $f_c \frac{K-1}{K}$

Number of output freqs

$= N = K^S$

$S = \#$ Stages

Output Spacing $\Delta f = \delta f \frac{K}{N}$

or $\delta f = K^{S-1} \Delta f$

Figure A.22

DIGITAL CONTROLLED FREQ SYNTHESIS (ITERATED

DIVIDE AND ADD)

FREQ SYNTHESIS - EXAMPLE

K = # base freq.

δf = base freq. separation

$N = K^s$

f_c = center freq.

$\delta f = K^{s-1} \Delta f$

If K = 2, s = 3 then $\delta f = 4\Delta f$ and $N = 2^3 = 8$

Suppose $\Delta f = 1$, then $\delta f = 4$

Suppose $f_c = 10$

Then choose base freqs s.t.

$$\frac{1}{K} \sum_{i=1}^{K} f_i = \frac{10}{2}$$

OR

$$f_1 = 3 , \quad f_2 = 7$$

$$(\delta f = 4)$$

STAGE 1

INPUT(S)	CONTROL)	ADD	÷2
5	0	3 = 8	4
	1	7 = 12	6

STAGE 2

INPUT(S)	CONTROL ↓	ADD	÷2
4	00	3 = 7	3.5
4	01	7 = 11	5.5
6	10	3 = 9	4.5
6	11	7 = 13	6.5

STAGE 3

INPUT(S)	CONTROL ↓	ADD
3.5	000	3 = 6.5
3.5	001	7 = 10.5
5.5	010	3 = 8.5
5.5	011	7 = 12.5
4.5	100	3 = 7.5
4.5	101	7 = 11.5
6.5	110	3 = 9.5
6.5	111	7 = 13.5

6.5	7.5	8.5	9.5	$f_c=10$ ↓	10.5	11.5	12.5	13.5
000	100	010	110		001	101	011	111

$$\Delta f = 1$$
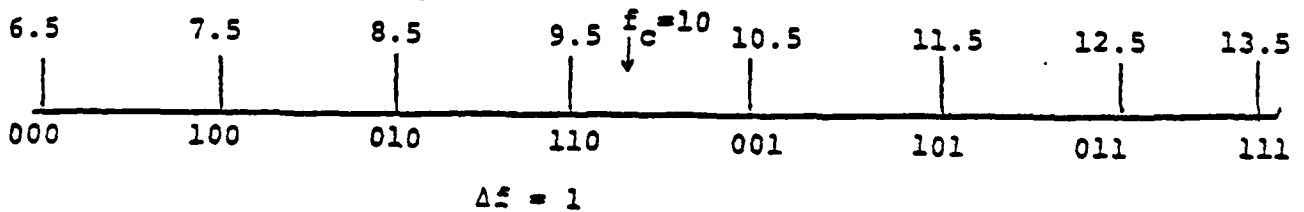
DIRECT SYNTHESIS ADVANTAGES

- Fast switching (0.1u sec) and settling time
- Extremely fine resolution (10^{-2} Hz or better)
- Good spurious rejection (60 - 100 dB)
- Low phase jitter
- Wide bandwidth ($>10^2$ MHz)
- Digital control
- Modular

DISADVANTAGES

- Limited percentage BW
- Large number parts
- Cost

A.12 Frequency Hopping Acquisition and Tracking

One of the principal reasons for using frequency hopping
as a means of spread spectrum is due to the fact that very rapid
acquisition is achievable without sacrificing processing gain
during the acquisition process. Perhaps the simplest acquisition
process to examine is the sliding sequential search similar to
that used in PN sequence acquisition. In frequency hopping,
the autocorrelation function width (in seconds) depends only on
the hopping rate (see figure A.12) as it does in direct sequence
signals. Unlike direct sequence signals however, the processing
gain is not determined by the hopping rate but rather by the
number of frequencies used in the pseudorandom hop pattern.
This allows us the flexibility of having a slow hop with broad
autocorrelation function and consequent broad timing tolerance for

- 223 -

rapid acquisition, while still permitting full and arbitrary processing gain to be achieved by using a sufficient number of frequencies. The acquisition time is given by

$$T_{acq} = \frac{k}{R} \frac{sec}{chip} \times \frac{1}{T} \frac{chips}{sec} \times \delta t$$

where k = # bits/chip searched

R = # bits/sec (data rate)

T = duration of a chip (sec)

δt = timing uncertainly (sec)

The expression for acquisition time assumes that either we have previously acquired and have lost signal so that reacquisition is attempted with an accumulated timing uncertainly of δt seconds or else that some prearranged preamble begins the acquisition sequence but an uncertainty of δt seconds is introduced in the process. The acquisition time is simply the time it takes to search the δt seconds. In terms of chips this is simply $\frac{\delta t}{T}$ chips

in order to integrate over the data rate so that full correlation processing gain is achieved, we must spend $\frac{k}{R}$ seconds for each chip that is searched. As an example for fast hopping, suppose k = 3, R = 1KBPS and T = 1μ sec (hop rate of 1 MHz). Suppose that clock stability is 0.1 ppm (10^{-7}) and communication is disruptedfor 10,000 seconds (<3 hrs) then $\delta t = 10^{-7} \times 10^4 = 1$ ms. Then the clock drift would have accumulated at most $\frac{1ms}{1\mu s} = 1000$ chips. Since we must spend 3 ms/chip the acquisition time is at most 3 seconds. For the same example, if we needed a processing gain of 40 dB we woudl use 10^4 frequencies of a total bandwidth of 10 MHz. If we needed 50 dB we would use 10^5 frequencies, etc.

- Constant envelope r.f. signal makes linear power ampliers unnecessary

- Near ÷ far ratio tolerable = 10^{-1}
  (= 20 dB loss for line of sight radio link)

- Most jammers equivalent to noise

- Timing accuracy needed <0.1 B_{RF}^{-1}

- Frequency accuracy needed <0.1 R (R is data bank rate)

- Slow acquisition unless parallel processing on matched filters are used

- $T_{acq} = 3 \times (P.G.) \times \delta t$ (δt is timing uncertainty)

- Effective against "look-thru" repeater provided codes are long

- High spurious responses when partial period correlation takes place or when data rate is integrally related to code repetition rate

- Code imbalance and/or equipment misalignment can obliterate anti-jam protection

Frequency Hopping

- Almost arbitrary band spreading is possible. Can cover entire technology band. Ex. B_{RF} = 1.5 GHZ at 10 GHZ with 1500 frequencies spaced 7KHZ apart.

- Processing gain is independent of hopping rate

- Effectively an interference avoidance scheme rather than an interference averaging system

- No code imbalance or equipment alignment problems

- Constant envelope signal

The acquisition time would still be 5 seconds. By contrast, if a direct sequence system were used that required 40 dB of processing gain with the same information rate (1kBPS) then a PN chip rate of 10^7 Hz would be needed. Using a serial search over 1 ms requires that we search through 1 ms/0.1us = 10,000 PN chips. At 3 ms/chip this would require 30 seconds. For slow frequency hopping, the result is even more dramatic. For example if the hop rate was once per millisecond then the acquisition time for the frequency hopping system would be 5 ms! Slow hopping has the disadvantage of being vulnerable to repeater jamming. Frequency hopping acquisition also lends itself to even further improvement in acquisition time (as does PN direct sequence spread spectrum) by the use of multiple processors and matched filters.

A.13 Comparison of PN Direct Sequence and Frequency Hopping

We conclude this appendix on frequency hopping by a brief point-by-point comparison of the benefits of frequency hopping and PN direct sequence spread spectrum modulation.

Pseudo-Noise

- Binary shift register generators

- Current-day limit B_{RF} = 100 - 400 MHz

- Effective against multipath greater than 1 chip

- Suitable for accurate ranging and timing

- Selective addressing possible for multiple access

- Partial band jammer may be effective unless coding is employed

- Lends itself well to nonbinary coding which is effective against burst errors

- Selective addressing possible

- Timing accuracy required is $<0.1\ T_c$ (T_c is the chip inferred, seconds)

- Frequency accuracy required is $<0.1\ R$

- Near $\div$ far ratio tolerable $= 10^{-3}$

- Slow hop rates easy to achieve and permit very rapid acquisition

- $T_{acq} = 10\ R^{-1}$ (milliseconds)

- Not suitable for accurate ranging and timing

- Fast hopping requires an expensive digitally controlled frequency synthesiser

- May be used as a preamble for fast acquisition of PN system or in hybrid configuration